3rd Conference on Language, Data and Knowledge

LDK 2021, September 1–3, 2021, Zaragoza, Spain

Edited by

Dagmar Gromann
Gilles Sérasset
Thierry Declerck
John P. McCrae
Jorge Gracia
Julia Bosque-Gil
Fernando Bobillo
Barbara Heinisch

Editors

Dagmar Gromann
University of Vienna, Austria
dagmar.gromann@gmail.com

Thierry Declerck
DFKI GmbH, Germany
declerck@dfki.de

Jorge Gracia
University of Zaragoza, Spain
jogracia@unizar.es

Fernando Bobillo
University of Zaragoza, Spain

Gilles Sérasset
Université Grenoble Alpes, France
gilles.serasset@imag.fr

John P. McCrae
National University of Ireland Galway, Ireland
john.mccrae@insight-centre.org

Julia Bosque-Gil
University of Zaragoza, Spain

Barbara Heinisch
University of Vienna, Austria
barbara.heinisch@univie.ac.at

ACM Classification 2012
Computing methodologies → Natural language processing; Computing methodologies → Knowledge representation and reasoning

ISBN 978-3-95977-199-3

Published online and open access by
Schloss Dagstuhl – Leibniz-Zentrum für Informatik GmbH, Dagstuhl Publishing, Saarbrücken/Wadern, Germany. Online available at https://www.dagstuhl.de/dagpub/978-3-95977-199-3.

Publication date
August, 2021

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at https://portal.dnb.de.

Digital Object Identifier: 10.4230/OASIcs.LDK.2021.0

ISBN 978-3-95977-199-3 **ISSN 1868-8969** **https://www.dagstuhl.de/oasics**

OASIcs – OpenAccess Series in Informatics

OASIcs is a series of high-quality conference proceedings across all fields in informatics. OASIcs volumes are published according to the principle of Open Access, i.e., they are available online and free of charge.

ISSN 1868-8969

https://www.dagstuhl.de/oasics

Contents

3rd Conference on Language, Data and Knowledge (LDK 2021).
Editors: Dagmar Gromann, Gilles Sérasset, Thierry Declerck, John P. McCrae, Jorge Gracia, Julia Bosque-Gil, Fernando Bobillo, and Barbara Heinisch

OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Knowledge Graphs

Applications for Language, Data and Knowledge

Use Cases in Language, Data and Knowledge

Preface

This volume presents the proceedings of the 3rd Conference on Language, Data and Knowledge (LDK 2021) held in Zaragoza, Spain from September 1–3, 2021. Language, Data and Knowledge is a biennial conference series on matters of human language technology, data science, and knowledge representation, initiated in 2017 by a consortium of researchers from the Insight Centre for Data Analytics at the National University of Ireland, Galway (Ireland), the Institut für Angewandte Informatik (InfAI) at the University of Leipzig (Germany), and the Applied Computational Linguistics Lab (ACoLi) at Goethe University Frankfurt am Main (Germany), and it has been supported by an international Scientific Committee of leading researchers in Natural Language Processing, Linked Data and Semantic Web, Language Resources and Digital Humanities. This initial conference was successfully continued in the second edition of LDK in Leipzig, Germany in 2019, organized by the Institut für Angewandte Informatik (InfAI) and co-organized by the Insight Centre for Data Analytics and the Applied Computational Linguistics Lab (ACoLi).

This third edition of the LDK conference is hosted by the University of Zaragoza in Zaragoza, Spain. Major support was provided by the NexusLinguarum COST Action CA18209 "European network for Web-centred linguistic data science", the Prêt-à-LLOD project funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 825182, and the University of Zaragoza.

As a biennial event, LDK aims at bringing together researchers from across disciplines concerned with the acquisition, curation and use of language data in the context of data science and knowledge-based applications. With the advent of the Web and digital technologies, an ever increasing amount of language data is now available across application areas and industry sectors, including social media, digital archives, company records, etc. The efficient and meaningful exploitation of this data in scientific and commercial innovation is at the core of data science research, employing NLP and machine learning methods as well as semantic technologies based on knowledge graphs.

Language data is of increasing importance to machine learning-based approaches in NLP, Linked Data and Semantic Web research and applications that depend on linguistic and semantic annotation with lexical, terminological and ontological resources, manual alignment across language or other human-assigned labels. The acquisition, provenance, representation, maintenance, usability, quality as well as legal, organizational and infrastructure aspects of language data are therefore rapidly becoming major areas of research that are at the focus of the conference.

Knowledge graphs is an active field of research concerned with the extraction, integration, maintenance and use of semantic representations of language data in combination with semantically or otherwise structured data, numerical data and multimodal data among others. Knowledge graph research builds on the exploitation and extension of lexical, terminological and ontological resources, information and knowledge extraction, entity linking, ontology learning, ontology alignment, semantic text similarity, Linked Data and other Semantic Web technologies. The construction and use of knowledge graphs from language data, possibly and ideally in the context of other types of data, is a further specific focus of the conference.

A further focus of the conference is the combined use and exploitation of language data and knowledge graphs in data science-based approaches to use cases in industry, including biomedical applications, as well as use cases in humanities and social sciences.

3rd Conference on Language, Data and Knowledge (LDK 2021).
Editors: Dagmar Gromann, Gilles Sérasset, Thierry Declerck, John P. McCrae, Jorge Gracia, Julia Bosque-Gil, Fernando Bobillo, and Barbara Heinisch
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

In total, 71 papers were submitted and reviewed by 69 reviewers. Typically, at least 3 reviews per paper resulted in 38 accepted papers for oral and poster presentations. As a novel feature, LDK 2021 had a special track for "Crazy New Ideas", that is, short abstracts that provide the occasion to present challenging research ideas that have not yet been fully explored or that the researchers would like to see in ten years from now. This category was decidedly aimed at research creativity and sparking interesting novel discussions within the LDK community. Three such crazy new ideas could be accepted for LDK and for publication.

Organizing Committee

Conference Chairs

John P. McCrae (National University of Ireland Galway, Ireland)
Thierry Declerck (DFKI GmbH, Germany)

Local Organizers

Julia Bosque Gil (University of Zaragoza, Spain)
Fernando Bobillo (University of Zaragoza, Spain)
Jorge Gracia (University of Zaragoza, Spain)

Program Chairs

Dagmar Gromann (University of Vienna, Austria)
Gilles Sérasset (Université Grenoble Alpes, France)

Workshop Chairs

Sara Carvalho (Universidade de Aveiro, Portugal)
Renato Rocha Souza (Austrian Academy of Sciences, Austria)

Proceedings Chair

Barbara Heinisch (University of Vienna, Austria)

3rd Conference on Language, Data and Knowledge (LDK 2021).
Editors: Dagmar Gromann, Gilles Sérasset, Thierry Declerck, John P. McCrae, Jorge Gracia, Julia Bosque-Gil, Fernando Bobillo, and Barbara Heinisch

OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Scientific Advisory Committee

John P. McCrae (National University of Ireland Galway, Ireland)
Paul Buitelaar (National University of Ireland Galway, Ireland)
Christian Chiarcos (Goethe-University Frankfurt, Germany)
Tatjana Gornostaja (Tilde, Latvia)
Philipp Cimiano (Bielefeld University, Germany)
Gerard de Melo (Rutgers University, USA)
Francis Bond (Nanyang Technological University, Singapore)
Thierry Declerck (DFKI GmbH, Germany)
Franciska de Jong (CLARIN ERIC, the Netherlands)
Karin Verspoor (University of Melbourne, Australia)
Edward Curry (National University of Ireland Galway, Ireland)
Jorge Gracia (University of Zaragoza, Spain)
Nancy Ide (Vassar College, USA)
Milan Dojchinovski (InfAI @ Leipzig University, Germany / CTU in Prague, Czech Republic)

3rd Conference on Language, Data and Knowledge (LDK 2021).
Editors: Dagmar Gromann, Gilles Sérasset, Thierry Declerck, John P. McCrae, Jorge Gracia, Julia Bosque-Gil, Fernando Bobillo, and Barbara Heinisch

OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Program Committee

Alessandro Adamou (The Open University)

Nathalie Aussenac-Gilles (IRIT CNRS)

Denilson Barbosa (University of Alberta)

Valerio Basile (University of Turin)

Pierpaolo Basile (University of Bari)

Martin Benjamin (Kamusi Project International)

Michael Bloodgood (The College of New Jersey)

Julia Bosque-Gil (Universidad de Zaragoza)

Paul Buitelaar (NUI Galway)

Harry Bunt (Tilburg University)

Aljoscha Burchardt (DFKI GmbH)

Eliot Bytyçi (University of Prishtina)

Nicoletta Calzolari (Istituto di Linguistica Computazionale – CNR)

Philipp Cimiano (Bielefeld University)

Gerard de Melo (HPI, University of Potsdam)

Thierry Declerck (DFKI GmbH)

Milan Dojchinovski (Czech Technical University in Prague)

Patrick Ernst (Amazon)

Maria Eskevich (CLARIN ERIC)

Luis Espinosa-Anke (Cardiff University)

Thierry Fontenelle (European Investment Bank)

Francesca Frontini (Istituto di Linguistica Computazionale A.Zampolli – CNR – Pisa)

Debanjan Ghosh (Educational Testing Service)

Hugo Gonçalo Oliveira (University of Coimbra)

Jeff Good (University at Buffalo)

Eero Hyvönen (Aalto University and University of Helsinki (HELDIG))

Nancy Ide (Vassar College)

Sepehr Janghorbani (Rutgers University)

Besim Kabashi (Friedrich-Alexander-Universität Erlangen-Nürnberg)

Te Taka Keegan (Waikato University)

Ilan Kernerman (K Dictionaries)

Dimitris Kontokostas (University of Leipzig)

Maria Koutraki (L3S Research Center, Leibniz University of Hannover)

Udo Kruschwitz (University of Regensburg)

Chaya Liebeskind (Jerusalem College of Technology, Lev Academic Center)

John P. McCrae (National University of Ireland Galway)

Margot Mieskes (University of Applied Sciences, Darmstadt)

Steven Moran (University of Neuchâtel)

Diego Moussallem (Paderborn University)

Alessandro Oltramari (Bosch Research and Technology Center)

Petya Osenova (Sofia University and IICT-BAS)

Bolette Pedersen (University of Copenhagen)

Laurette Pretorius (School of Interdisciplinary Research and Graduate Studies, University of South Africa)

Gábor Prószéky (MorphoLogic & PPKE)

Francesca Quattri (The Hong Kong Polytechnic University)

Alexandre Rademaker (IBM Research Brazil and EMAp/FGV)

3rd Conference on Language, Data and Knowledge (LDK 2021).
Editors: Dagmar Gromann, Gilles Sérasset, Thierry Declerck, John P. McCrae, Jorge Gracia, Julia Bosque-Gil, Fernando Bobillo, and Barbara Heinisch
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Simon Razniewski (Max Planck Institute for Informatics)

Georg Rehm (DFKI GmbH)

Nils Reiter (Institute of Natural Language Processing, Stuttgart University)

Steffen Remus (University of Hamburg)

Laurent Romary (INRIA & HUB-ISDL)

Mike Rosner (University of Malta)

Marco Rospocher (University of Verona)

Felix Sasaki (Cornelsen Verlag GmbH & TH Brandenburg)

Andrea Schalley (Karlstad University)

Max Silberztein (Université de Franche-Comté)

Steffen Staab (IPVS, Universität Stuttgart, DE and WAIS, University of Southampton, UK)

Armando Stellato (University of Rome, Tor Vergata)

Stan Szpakowicz (University of Ottawa)

Liling Tan (Nanyang Technological University)

Ciprian-Octavian Truica (Aarhus University)

Andrius Utka (Vytautas Magnus University)

Giedre Valunaite Oleskeviciene (Mykolas Romeris University)

Marieke van Erp (KNAW Humanities Cluster)

Marc Verhagen (Brandeis University)

Karin Verspoor (RMIT University)

Piek Vossen (Vrije Universiteit Amsterdam)

Qian Yang (Duke University)

Ziqi Zhang (Sheffield University)

The JeuxDeMots Project

Mathieu Lafourcade
LIRMM – Equipe TEXTE, University of Montpellier, France

Abstract

The JeuxDeMots project aims at building a very large knowledge base in French, both common sense and specialized, using games, contributory approaches, and inference mechanisms. A dozen games have been designed as part of this project, each one allowing to collect specific information, or to consolidate the information acquired through the other games. With this presentation, the data collected and constructed since the launch of the project in the summer of 2007 will be analyzed both qualitatively and quantitatively. In particular, the following aspects will be detailed: the structure of the lexical and semantic network, some types of relations (semantic, ontological, subjective, semantic roles, associations of ideas), annotation of relations (meta-information), semantic refinements (management of polysemy), the creation of clusters allowing the representation of richer knowledge (n-argument relations) that make an implicit neural network. Finally, I will describe some complementary acquisition methods and applications such as a bot for endogenous contributions, a chatbot making inferences and semantic extraction from texts.

2012 ACM Subject Classification Computing methodologies → Artificial intelligence; Computing methodologies → Natural language processing; Computing methodologies → Language resources

Keywords and phrases Lexical Semantic Network, Games with a Purpose, Inferences, Knowledge Representation, Semantic Representation

Digital Object Identifier 10.4230/OASIcs.LDK.2021.1

Category Invited Talk

3rd Conference on Language, Data and Knowledge (LDK 2021).
Editors: Dagmar Gromann, Gilles Sérasset, Thierry Declerck, John P. McCrae, Jorge Gracia, Julia Bosque-Gil, Fernando Bobillo, and Barbara Heinisch; Article No. 1; pp. 1:1–1:1

OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

A Smell is Worth a Thousand Words: Olfactory Information Extraction and Semantic Processing in a Multilingual Perspective

Sara Tonelli
Digital Humanities research Unit, Fondazione Bruno Kessler, Trento, Italy

Abstract

More than any other sense, smell is linked directly to our emotions and our memories. However, smells are intangible and very difficult to preserve, making it hard to effectively identify, consolidate, and promote the wide-ranging role scents and smelling have in our cultural heritage. While some novel approaches have been recently proposed to monitor so-called urban smellscapes and analyse the olfactory dimension of our environments (Quercia et al., [1]), when it comes to smellscapes from the past little research has been done to keep track of how places, events and people have been described from an olfactory perspective. Fortunately, some key prerequisites for addressing this problem are now in place. In recent years, European cultural heritage institutions have invested heavily in large-scale digitisation: we hold a wealth of object, text and image data which can now be analysed using artificial intelligence. What remains missing is a methodology for the extraction of scent-related information from large amounts of texts, as well as a broader awareness of the wealth of historical olfactory descriptions, experiences and memories contained within the heritage datasets. In this talk, I will describe ongoing activities towards this goal, focused on text mining and semantic processing of olfactory information. I will present the general framework designed to annotate smell events in documents, and some preliminary results on information extraction approaches in a multilingual scenario. I will discuss the main findings and the challenges related to modelling textual descriptions of smells, including the metaphorical use of smell-related terms and the well-known limitations of smell vocabulary in European languages compared to other senses.

2012 ACM Subject Classification Applied computing → Document analysis; Information systems → Digital libraries and archives

Keywords and phrases olfactory information extraction, smellscapes, multilingual annotation

Digital Object Identifier 10.4230/OASIcs.LDK.2021.2

Category Invited Talk

References

1 Daniele Quercia, Rossano Schifanella, Luca Maria Aiello, and Kate McLean. Smelly maps: the digital life of urban smellscapes. In *Ninth International AAAI Conference on Web and Social Media*, 2015.

3rd Conference on Language, Data and Knowledge (LDK 2021).
Editors: Dagmar Gromann, Gilles Sérasset, Thierry Declerck, John P. McCrae, Jorge Gracia, Julia Bosque-Gil, Fernando Bobillo, and Barbara Heinisch; Article No. 2; pp. 2:1–2:1

OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Free/Open-Source Machine Translation for the Low-Resource Languages of Spain

Mikel L. Forcada
Departament de Llenguatges i Sistemes Informàtics,
Universitat d'Alacant, 03690 Sant Vicent del Raspeig, Spain
Prompsit Language Engineering, 03202 Elx, Spain

Abstract

While machine translation has historically been rule-based, that is, based on dictionaries and rules written by experts, most present-day machine translation is corpus-based. In the last few years, statistical machine translation, the dominant corpus-based approach, has been displaced by neural machine translation in most applications, in view of the better results reported, particularly for languages with very different syntax. But both statistical and neural machine translation need to be trained on large amounts of parallel data, that is, sentences in one language carefully paired with their translations in their other language, and this is a resource that may not be available for some low-resource languages. While some of the languages of Spain may be considered to be reasonably endowed with parallel corpora connecting them to Spanish or even to English – Basque, Catalan, Galician –, and are well-served with machine translation systems, there are many other languages which cannot afford them such as Aranese Occitan, Aragonese, or Asturian/Leonese. Fortunately, languages in this last group belong to the Romance language family, as Spanish does, and this makes translation from and into Spanish under a rule-based paradigm the only feasible approach. After describing briefly the main machine translation paradigms, I will describe the Apertium free/open-source rule-based machine translation platform, which has been used to build machine translation systems for these low-resource languages of Spain, indeed, sometimes the only ones available. The free/open-source setting has made linguistic data for these languages available for anyone in their linguistic communities to build other linguistic technologies for these low-resourced languages. For example, the Apertium family of bilingual and monolingual data has been converted into RDF and they have been made accessible on the Web as linked data.

2012 ACM Subject Classification Applied computing → Language translation

Keywords and phrases free/open-source, machine translation, languages of Spain, low-resource machine translation

Digital Object Identifier 10.4230/OASIcs.LDK.2021.3

Category Invited Talk

3rd Conference on Language, Data and Knowledge (LDK 2021).
Editors: Dagmar Gromann, Gilles Sérasset, Thierry Declerck, John P. McCrae, Jorge Gracia, Julia Bosque-Gil, Fernando Bobillo, and Barbara Heinisch; Article No. 3; pp. 3:1–3:1
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

A Computational Simulation of Children's Language Acquisition

Ben Ambridge
ESRC International Centre for Language and Communicative Development (LuCiD), University of Liverpool, UK

Abstract

Many modern NLP models are already close to simulating children's language acquisition; the main thing they currently lack is a "real world" representation of semantics that allows them to map from form to meaning and vice-versa. The aim of this "Crazy Idea" is to spark a discussion about how we might get there.

2012 ACM Subject Classification Theory of computation → Grammars and context-free languages

Keywords and phrases Child language acquisition, language development, deep learning, BERT, ELMo, GPT-3

Digital Object Identifier 10.4230/OASIcs.LDK.2021.4

Category Crazy New Idea

Funding This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant Agreement No. 681296: CLASS). Ben Ambridge is a Professor in the International Centre for Language and Communicative Development (LuCiD) at The University of Liverpool. The support of the Economic and Social Research Council [ES/L008955/1] is gratefully acknowledged.

1 Crazy Idea

Modern NLP systems such as BERT, ELMo and GPT-3 have many potential applications in both industry and academia; but one that has barely been considered is simulating how children learn their native language. This question lies at the very heart of cognitive science – with at least five journals devoted solely to it – but has yet to be tackled with modern NLP approaches. Although modelling work is conducted in this domain, it typically uses small and simple models (e.g., three-layer connectionist networks) to tackle narrowly circumscribed problems (for example, children's acquisition of the English past-tense system; [10]).

But here's the thing: Models like BERT, ELMo and GPT-3 are, in many respects, nearly there. What the past 50 years of child language research have taught us is that learners store representations at every level from the concrete (e.g., the lexical string *cup+of+tea*) to the abstract (e.g., the *SUBJECT VERB OBJECT* transitive construction), and everything in between (see [1, 2] for reviews). That is, exemplars – utterances that children hear and store – are never discarded in favour of context-free symbolic rules. Rather these exemplars are re-represented at increasingly abstract levels, just as in BERT, ELMo, GPT-3 (and other deep-learning models in domains such as image-classification models; e.g., [14]).

Crucially, as I argued in [2], "this type of model is not just a metaphor ([5, 6, 9]). The brain really does contain multiple layers of units (i.e., neurons), each of which aggregates input signals using a nonlinear function and outputs signals to other units. While any particular artificial neural network model of language is only the clumsiest metaphor, the claim that language is represented as patterns of activation across 'dumb' neurons, each of which 'knows' nothing about nouns, verbs and all the rest of it is literally true, and quite beyond dispute".

So what's missing? Why aren't BERT and the like already viable candidates for models of children's language acquisition? The answer, of course, is that BERT lacks not only any kind of communicative goals, but any links to real-world meanings at all (e.g., [3, 11, 12]), with "meanings" represented solely as contextualized word embeddings. What we need, then, is a deep-learning model that learns like children; a model that – when "listening" – maps strings onto meanings and – when speaking – maps "meanings" onto strings.

Of course, this type of approach was tried in the earliest days of NLP, and swiftly abandoned as unworkable. And, indeed, if our goal is to translate from one natural language to another, to develop a predictive-text application, or to generate passages of text given a prompt (e.g., GPT-3), contextualized word embeddings will probably do a better job. But if our goal is to simulate children's language acquisition, we have to bite the bullet and develop "real-world" semantic representations (which, as Gary Marcus has often argued, are important for many practical applications of NLP too).

Indeed, simulating the first few years of language acquisition may be a useful way to take the first steps towards this much bigger problem. A typical two-year-old has a vocabulary of only a couple of hundred words; a typical three-year-old, a couple of thousand. The amount of input that children receive – around 10,000–20,000 words of speech per day – is also small by BERT standards, and most of it is relatively simple, concrete and highly repetitive ([4]). Thus, simulating the first few years of child language acquisition in its entirety is a realistic goal for an ambitious and well-funded research team, even if some hand coding of semantics is required (though a wrinkle here is the extent to which children's semantic representations are adultlike).

How should we go about this problem? This is where I hand over to you (and why I'm submitting this paper as a "Crazy Idea" for discussion). I'm a child language experimentalist, with only very limited experience of basic computational modelling. Could knowledge graphs represent the necessary semantic information? Would we need some additional hand-coding of at least the basic objects and actions in the child's world? And, if so, can we adopt a "view from nowhere", or do we need to take account of the fact that human cognition is embodied in our sensorimotor experience ([8]), perhaps by including something like sensorimotor norms (e.g., [7])?

Could neural-symbolic approaches (e.g., [13]) connect knowledge graphs with neural networks? And what other semantic representations are used in modern NLP? Or can we use some kind of vector representation after all, perhaps using principal component analysis to distil them into elements of meaning that can be used to encode semantic "messages". Or can we somehow represent meaning by leveraging techniques used in machine translation and using crosslinguistic vectors (e.g., the "meaning" of *cat* is the entity that stands in the same relationship to *dog* as does French *chat* to *chien*)? You tell me!

References

1 Ben Ambridge. Against stored abstractions: A radical exemplar model of language acquisition. *First Language*, 40(5-6):509–559, 2020a.

2 Ben Ambridge. Abstractions made of exemplars or "you're all right, and i've changed my mind": Response to commentators. *First Language*, 40(5-6):640–659, 2020b.

3 Emily M Bender and Alexander Koller. Climbing towards nlu: On meaning, form, and understanding in the age of data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5185–5198, 2020.

4 Thea Cameron-Faulkner, Elena Lieven, and Michael Tomasello. A construction based analysis of child directed speech. *Cognitive Science*, 27(6):843–873, 2003.

5 Daniel C Dennett. *From bacteria to Bach and back: The evolution of minds*. WW Norton & Company, 2017.

6 Uri Hasson, Samuel A Nastase, and Ariel Goldstein. Direct fit to nature: An evolutionary perspective on biological and artificial neural networks. *Neuron*, 105(3):416–434, 2020.

7 Dermot Lynott, Louise Connell, Marc Brysbaert, James Brand, and James Carney. The lancaster sensorimotor norms: multidimensional measures of perceptual and action strength for 40,000 english words. *Behavior Research Methods*, pages 1–21, 2019.

8 Jean M Mandler. How to build a baby: Ii. conceptual primitives. *Psychological Review*, 99(4):587, 1992.

9 Andrea E Martin. A compositional neural architecture for language. *Journal of Cognitive Neuroscience*, 32(8):1407–1427, 2020.

10 James L McClelland, David E Rumelhart, PDP Research Group, et al. *Parallel distributed processing*, volume 2. MIT press Cambridge, MA, 1986.

11 William Merrill, Yoav Goldberg, Roy Schwartz, and Noah A. Smith. Provable limitations of acquiring meaning from ungrounded form: What will future language models understand? *CoRR*, abs/2104.10809, 2021. `arXiv:2104.10809`.

12 Mitja Nikolaus and Abdellah Fourtassi. Evaluating the acquisition of semantic knowledge from cross-situational learning in artificial neural networks, May 2021. `doi:10.31234/osf.io/mbesf`.

13 Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Joshua B Tenenbaum. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. *arXiv preprint*, 2018. `arXiv:1810.02338`.

14 Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization, 2017. `arXiv:1611.03530`.

Get! Mimetypes! Right!

Christian Chiarcos
Applied Computational Linguistics Lab, Goethe University Frankfurt, Germany

Abstract

This paper identifies three technical requirements – availability of data, sustainable hosting and resolvable URIs for hosted data – as minimal pre-conditions for Linguistic Linked Open Data technology to develop towards a mature technological ecosystem that third party applications can build upon. While a critical amount of data is available (and it continues to grow), there does not seem to exist a hosting solution that combines the prospects of long-term availability with an unrestricted capability to support resolvable URIs. In particular, data hosting services do currently not allow data to be declared as RDF content by means of their media type (mime type), so that the capability of clients to recognize formats and to resolve URIs on that basis is severely limited.

2012 ACM Subject Classification Applied computing → Format and notation; Applied computing → Document management and text processing; Software and its engineering → Interoperability

Keywords and phrases data hosting, mimetypes, resolvability, URIs, Linked Data foundations

Digital Object Identifier 10.4230/OASIcs.LDK.2021.5

Category Crazy New Idea

Linked (Open) Data is generally considered to be the prototypical technology to implement the "FAIR Guiding Principles for scientific data management and stewardship" [4], and for the specific domain of language resources, Linguistic Linked Open Data (LLOD) comes with the promise to facilitate the integration of linguistic information from and across distributed resources. In the more general context of web technology, this represents probably the most promising way to address challenges of multilinguality – and indeed, since the publication of the OntoLex-Lemon vocabulary in 2016,[1] this is specifically the area where technology and data are most mature, and LLOD is on the verge of becoming a mainstream technology.

The success of the LLOD vision, however, crucially depends on finding solutions for three elementary problems:[2]

(1) It is necessary to provide a critical amount of data in RDF and with links,
(2) it is necessary to provide sustainable hosting solutions so that future applications can rely on the availability of a resource,
(3) it is necessary to provide resolvable URIs for the data such that it can be addressed as Linked Data and used as such.

The community has gone a long way since the inception of the LLOD cloud in 2010 [3], and especially in the lexical domain, the first challenge is basically overcome. Several established sub-communities in the field now support LLOD technology, e.g., the WordNet community [1], and massive amounts of OntoLex-compliant lexical data are available, covering more than 400 language varieties with substantial dictionaries, e.g., [2].

[1] `https://www.w3.org/2016/05/ontolex/`
[2] These are not the only aspects where LLOD technology suffers from bottle-necks. Problems also exist when it comes to tooling, ease of use, the challenges to develop agreed-upon vocabularies to exploit possible synergies, etc. However, these are challenges on the user side, and they can be addressed if researchers, users, providers and engineers devote time and energy. The problems mentioned here are more elementary in that they are necessary to constitute a technical environment to publish, access and maintain previously created data. Without hope for arriving at such an ecosystem within a reasonable time frame, enthusiasm, time and energy will be invested in vain and quickly decay.

3rd Conference on Language, Data and Knowledge (LDK 2021).
Editors: Dagmar Gromann, Gilles Sérasset, Thierry Declerck, John P. McCrae, Jorge Gracia, Julia Bosque-Gil, Fernando Bobillo, and Barbara Heinisch; Article No. 5; pp. 5:1–5:4

OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

As for the second challenge, the publication of data and its maintenance for subsequent use, replication and verification has been a problem for academic research in general. This is mostly caused by the fact that data is often produced in the context of temporary investments, e.g., as part of thesis projects or research grants. Traditionally, neither addressed questions of long-term data storage: A student will not have the resources and simply move on to other challenges after accomplishing a degree, and a fixed-term research project will eventually run out of funding. Publication via web sites may work for some time, but as soon as the local IT department or the hosting institution undergoes any form of major restructuring, much data is likely to be relocated – if not lost. So, unless designated efforts for preservation and link updates are being made, the life expectancy of a legacy dataset published in this way is at maybe, around 5 years after the project finished. Libraries may help here, but then, policies with respect to data hosting differ greatly, publications will always take priority in this context, much more so than data hosting, and resources are severely limited, e.g., in size of data dumps permitted.

Luckily, things improved greatly in this respect. With platforms such as Zenodo,[3] researchers now have the possibility to deposit their data under a persistent URL, with the large number of CLARIN centers established in the last decade,[4] there are regional solutions specifically for tools and data from natural language processing and the language sciences over all the EU, and with the growth of DARIAH[5] and SSHOC,[6] comparable ecosystems also emerge in the Humanities and Social Sciences. But even independently from designated research funding, established commercial solutions do exist which may depend to a lesser degree from central funding, e.g., GitHub[7] is occasionally used for the purpose. So, the challenge of data hosting has also been largely overcome.

At the moment, a major bottle-neck for LLOD technology is the third challenge. Open source RDF data will only be able to become Linked Open Data if the individual data points ("things") can be addressed with resolvable URIs. Looking at Zenodo as an example, it is possible to deposit RDF data, of course. And if that data uses persistent URIs created by a redirection service such as W3ID or Purl, it is possible to redirect them to the specific URL/DOI generated by Zenodo. So, they could resolve, in theory.

The problem here is that they can resolve only if the data is recognized as RDF data by an application that accesses the data dump. The standard way for doing so would be by an RDF mimetype such as text/turtle (etc., for other RDF formats). Unfortunately, the mimetype of data in Zenodo is either application/octet-stream or text/plain.

This means that applications need to guess the format if they attempt to resolve URIs against a resource. This can work, but it is unreliable. In particular, it will fail if URIs do not include the file ending (as recommended, because we have content negotiation, except not here), or if the data URI carries any flags after the file ending (e.g., "...?download=1").

Let's take the Jena-based service under `http://www.sparql.org` as an example, with a short query against `https://zenodo.org/record/4444132/files/crmtex.owl?download=1`:

```
SELECT *
WHERE { ?a ?b ?c } LIMIT 10
```

[3] https://zenodo.org/
[4] https://www.clarin.eu/
[5] https://www.dariah.eu/
[6] https://www.sshopencloud.eu/
[7] https://github.com/

The service of `sparql.org` does allow to query RDF data on the web, without the need to set up a local SPARQL end point or to download any data, so this is a nice demonstrator for RDF-based web services. Moreover, the SPARQL query can be added to the URI, so, it can be re-purposed in other web services and, for example, consulted via the `LOAD` keyword from a local SPARQL end point. With minimal effort, this web service is capable to demonstrate the key benefits of federation and information integration without putting the burden to maintain or set up any infrastructure on the developer of a particular query.[8]

Unfortunately, this fails with the original Zenodo data link.[9] In this case, it will work if flags are stripped and the file extension is recognized,[10] but this not robust (it is guesswork specific to this particular implementation and not guaranteed to work with other consumers). In essence, while the SPARQL query is portable and due to the use of W3C standards, the data is, as well, the behavior of your local triple store is somewhat unpredictable. Depending on specific heuristics to determine the content of the RDF data, it will perform differently (if at all).

The problem is not limited to the `FROM` keyword: With your local triple store, you might want to use the `LOAD` keyword of SPARQL, for example, to retrieve a remote data set. But again, the same problem arises if the mediatype of the data to be loaded from a remote host is not declared. Furthermore, the problem is not specific to Zenodo, it is only an example. In fact, I am not aware of any provider of LOD-compliant hosting services for an unrestricted pool of data providers. To illustrate a real-world example involving a commercial provider, GitHub displays its "raw" data similarly as text/plain. For example, the persistent URL `http://purl.org/acoli/conll` redirects to `https://raw.githubusercontent.com/acoli-repo/conll-rdf/master/owl/conll.ttl`, but this is exposed as text/plain, not text/turtle. Whether or not a particular SPARQL engine will be able to resolve this URI (note that – in accordance with LOD best practices –, the persistent URI does not include the file extension!) will vary across different implementations, giving the entire technology the appeal to be fragile and unreliable.

Fixing this by supporting RDF-compliant media types could unleash a wave of new demonstrators of the technology, that illustrate data re-use and integration from Zenodo and other portals. As it stands, these demonstrators often run against unstable university pages – or just quietly break. Having them run against data dumps hosted at Zenodo or other academic data maintainers would guarantee the necessary longevity to reliably demonstrate federated search to students, scholars and future generations.

Indeed, complementing existing hosting services with LOD-compliant, resolvable URIs would establish the minimal technical level of interoperability required to make existing (L)LOD data and services stable, sustainable and eventually operational. Moreover, reliable long-term hosting would enable commercial use cases. At the moment, the lack of confidence in long-term availability of LOD data sets represents a bias for the development of applications and services that depend on any such data. But only if Linked Data also works in a business context (and the potential is great), its vision and prospects will be able to unfold.

---

[8] In comparison to a local triple store there is a limitation in performance and scalability. But it is still an ideal, almost effortless environment for testing and demonstration.

[9] The following URI contains the corresponding query and the FROM clause points to the respective data source. The URI should resolve against a dynamically created query result. `http://www.sparql.org/sparql?query=SELECT+*FROM+<https://zenodo.org/record/4444132/files/crmtex.owl?download=1>WHERE+{+?a+?b+?c+}+LIMIT+10&default-graph-uri=&output=xml&stylesheet=/xml-to-html.xsl`.

[10] `http://www.sparql.org/sparql?query=SELECT+*FROM+<https://zenodo.org/record/4444132/files/crmtex.owl>WHERE+{+?a+?b+?c+}+LIMIT+10&default-graph-uri=&output=xml&stylesheet=/xml-to-html.xsl`

Overall, this is very easy to fix, and here comes a Crazy New Idea: Make a coordinated effort as a community to get providers of language resource infrastructures to support Linked Data compliant media types, e.g., petition repeatedly and massively to maintainers and developers of such infrastructures that data is declarable as text/turtle (etc.) than just text/plain or application/binary. After all, their decision to not support LOD-compliant mediatypes is a deliberate one, and it's not resulting from ignorance, but from a (somewhat lazy) risk-gain calculation: Data provided by a hosting service can be used to infuse malicious code into applications of clients, especially if it is automatically executed in the browser, and minimizing the number of supported mediatypes reduces this risk for the host, or better, it transfers the responsibility for executing malicious code from the host to the client. Given the current state of affairs, it is up to the providers and users of (Linguistic) Linked (Open) Data to explore that risk and to convince infrastructure providers that this risk is minimal (text/turtle is not interpreted by browsers, these days), that there is a potential gain for them (more functionalities, more popularity) and that there is a concrete need in their user community. Given the continued – and rising – popularity of Linguistic Linked Open Data, this point can be easily made, and – with the Cost Action Nexus Linguarum and several large-scale European and national projects based on this technology at this time – more easily so for language resources than for Linked Data in general.

It would be an exaggeration to call the idea to implement established standards crazy or even particularly innovative, but there is a new aspect I would like to throw into the discussion, that is, to address this technical problem also at the political level: Let's *collectively* approach infrastructure providers.

References

1 Francis Bond, Piek Vossen, John McCrae, and Christiane Fellbaum. CILI: The Collaborative InterLingual Index. In *Proc. of the 8th Global Wordnet Conference (GWC 2016)*, Bucharest, Romania, 2016.

2 Christian Chiarcos, Christian Fäth, and Maxim Ionov. The acoli dictionary graph. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 3281–3290, 2020.

3 Christian Chiarcos, John McCrae, Philipp Cimiano, and Christiane Fellbaum. Towards open data for linguistics: Linguistic linked data. In Alessandro Oltramari, Piek Vossen, Lu Qin, and Eduard Hovy, editors, *New Trends of Research in Ontologies and Lexical Resources: Ideas, Projects, Systems*, pages 7–25. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. doi:10.1007/978-3-642-31782-8_2.

4 Mark D. Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E. Bourne, Jildau Bouwman, Anthony J. Brookes, Tim Clark, Mercé Crosas, Ingrid Dillo, Olivier Dumon, Scott Edmunds, Chris T. Evelo, Richard Finkers, Alejandra Gonzalez-Beltran, Alasdair J.G. Gray, Paul Groth, Carole Goble, Jeffrey S. Grethe, Jaap Heringa, Peter A.C 't Hoen, Rob Hooft, Tobias Kuhn, Ruben Kok, Joost Kok, Scott J. Lusher, Maryann E. Martone, Albert Mons, Abel L. Packer, Bengt Persson, Philippe Rocca-Serra, Marco Roos, Rene van Schaik, Susanna-Assunta Sansone, Erik Schultes, Thierry Sengstag, Ted Slater, George Strawn, Morris A. Swertz, Mark Thompson, Johan van der Lei, Erik van Mulligen, Jan Velterop, Andra Waagmeester, Peter Wittenburg, Katherine Wolstencroft, Jun Zhao, and Barend Mons. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*, 3(1), 2016. doi:10.1038/sdata.2016.18.

Mind the Gap: Language Data, Their Producers, and the Scientific Process

Tobias Weber
Ludwig-Maximilians-Universität München, Germany

Abstract

This paper discusses the role of low-resource languages in NLP through the lens of different stakeholders. It argues that the current "consumerist approach" to language data reinforces a vicious circle which increases the technological exclusion of minority communities. Researchers' decisions directly affect these processes to the detriment of minorities and practitioners engaging in language work in these communities. In line with the conference topic, the paper concludes with strategies and prerequisites for creating a positive feedback loop in our research benefiting language work within the next decade.

2012 ACM Subject Classification Computing methodologies → Language resources; Computing methodologies → Natural language processing; Social and professional topics → Cultural characteristics; Software and its engineering → Software creation and management; Applied computing → Language translation

Keywords and phrases minority languages, data integration, sociology of technology, documentary linguistics, exclusion

Digital Object Identifier 10.4230/OASIcs.LDK.2021.6

Category Crazy New Idea

1 Introduction

This paper was inspired by the conference organisers' call for "challenging research ideas that [...] you would like to see in ten years from now". One of the major challenges associated to the conference topics is the relationship between different agents: individual speakers providing data, researchers annotating data, computational linguists and data scientists building applications on these data, and the public, i.e., speaking communities, using these applications. What seems like a mono-directional and circular relationship on the macro-level becomes a complex network of interactions on the micro-level. The discussion here shall be led with an interdisciplinary view to present these interactions and the role of three stakeholders: the speakers, the linguists, and the computer scientists. These three groups are to be seen as functional roles which are not mutually exclusive – there can be linguists and computer scientists who also qualify as speakers, or even individuals fulfilling all three roles simultaneously. The paper is written from the intermediary position of the linguist in this list, as it contains my personal experiences as a philologist and curator of language data. While my perspective may not be representative of all linguists, there is a consensus on professional standards in data collection and preparation. Especially for minority languages and recently documented languoids [11], researchers aim to generate sustainable and interoperable data sets for a variety of subsequent uses. As this topic is central to the scientific discourse in documentary linguistics, there is a bulk of literature addressing the issues in language data use. Although the focus of this paper shall be on future developments, it appears imperative to point out that already twenty years ago, before Big Data became a standard paradigm in computer science, linguists highlighted potential conflicts between this approach and the reality faced by those documenting endangered languages. As a consequence, some of these issues have not been fully solved or have worsened over the last decade, making their

solution a priority for the next decade. It is important to emphasise that the goal is not simply to force Western paradigms and conceptualisations onto endangered languages, as linguists and computer scientists have warned [8, 22]. The main principles must be the acknowledgement of community agency, reciprocity, and social awareness for the ways our work affects communities. We should not always seek the easy route, where we find large amounts of standardised data, but pay attention to calls for support with data collection and curation.

2 Standardisation – a necessary evil?

In their training, most linguists will learn about technical requirements for data they collect and aim to analyse in their work. This may include file types, encodings, methods of annotation, or tools for the creation of data sets. In many cases, there are accepted standards or conventions for the discipline of linguistics which often follow recommendations from computer scientists. And while these standards help to improve interoperability, durability, and usability in most instances, it would be a fallacy to regard standardisation as a solution in every scenario. Sometimes researchers inherit data sets or projects from colleagues, want to adhere to traditions in their discipline, or must respect requests by a stakeholder. Requiring them to change standards and conventions may create challenges which can only be resolved by investing time and money. In any case, linguists need to be aware of the necessary standards and tools for ensuring compliance with these standards.

I agree with one reviewer that we need to be careful about seeing technology as the central solution in language documentation, where the preoccupation with technological features and facts leads us away from the central role of human relationships and community involvement [14]. This framing commodifies language and data, and has to be seen critical [27]. Instead, we need to take the community and its members into consideration, e.g., under the six Rs presented by Galla and Goodwill [17]: respect, relationality, relevance, responsibility, reciprocity, and resiliency. Yet, even this focus in contemporary documentation and revitalisation efforts will not remove the undesired notion of the commodity from language and data – it has been introduced through globalisation and the subsequent exposure to Western conceptualisations of language and can be found in several indigenous communities around the globe. We must not presume the Western interpretation [8] but, likewise, cannot deny its influence in some communities.

Major challenges are posed by instances where a conversion to a standard cannot be automated, e.g., if annotation, translation, or transcription layers need to be added or altered. These are crucial for many applications in NLP and their quality directly affects the usability of products sold or gifted to the communities. In these cases, data sets need to be manually curated, if the quality of application shall not be impaired. The recent trend of building applications using machine learning or big data approaches requires large amounts of data which needs to conform to particular standards, in addition to being comprehensive. For well-resourced languages (in terms of language resources, time, money, or skilled labour), these obstacles are overcome with seeming ease. Looking at the other end of the scale, under-resourced languages may suffer in multiple respect, relative to these standards. The accumulation of these issues can, subsequently, lead to exclusion, not least in a technological sense [18]. A multi-million token annotated corpus of "gold standard" is comparably rare considering the often cited figure of 7,000 languages and their numerous varieties. As we benchmark applications built for major languages as the state-of-the-art, we set high targets for under-resourced languages which they may not be able to achieve. As a result, we

further add to their exclusion. Although there are a range of approaches which aim to create applications from small or unstandardised data sets [2, 3, 5, 15, 16], these papers – in general – summon the "Zero Resource Scenario" criticised by Bird [8], while simultaneously removing speakers and, in some cases, documenters from the line of research. At the same time, these applications cannot be compared in coverage or reliability to applications for major languages and are not solutions against technological exclusion.

In terms of technological exclusion of a language community, we can approximate a trend for a given point in time or interval by calculating the ratio of increases in expected standards and requirements (i.e., size of corpora, quality of translations, consistency of annotation, variety of genres, balance of language data) to the increase in usability (quality and quantity) of language data for the particular languoid, which may include rate of documentation or addition to corpora; measures of quality, balance, or representativeness of data sets and their layers; and the adoption of standards. While these precise calculation of these measures or definition of requirements and standards depend on the desired quality and type of application, this yields $technological\ exclusion = \frac{increase\ in\ language\ resource\ usability}{increase\ in\ required\ standards}$. If the value is below 1, the community is increasingly technologically excluded; if it is at 1 or above the available data sets are sufficient for creating applications of the desired standard, with higher scores generally correlating to better quality and more diverse usage cases of the applications. If we use variables which measure absolute values (e.g., number of tokens in a corpus), we can even calculate how many words, translations, or annotations each speaker, annotator, or researcher must contribute $\frac{(required\ quantity - current\ state)}{community\ size}$. The point about the accumulative nature of exclusion becomes apparent if we accommodate for different capacities in producing language data (e.g., creating data through publications or social media) or ensuring quality (e.g., digitally available data, use of text processing tools). These factor into the equation on the side of the required standards, increasing the burden on each community member. The dynamic nature in standards and corpus development reinforces the trends in technological exclusion, making it more difficult or resource intensive to break the vicious circle. The availability of data conforming to the required standards creates a bottleneck in the creation of applications [1, 28], especially for communities which are already suffering from exclusion.

One reviewer questioned to what extent marginalised communities would want to use their languages in digital spaces. I cannot speak on the behalf of any community, yet, from my experience with European linguistic minorities, I know of positive responses to applications and tools for minority languages. On the one hand, we must not impose Western arguments about rationality, functionality, or instrumental value on these attitudes and decisions [8], as language use may be tied to other domains and reasons. On the other hand, technology falls together with media and telecommunication and forms a domain for language use which can also be a "marker of recognition in the digital realm" [8, p.3509]. In the study of minority language media, substitution with media in other languages is an important measure [24] which can be transferred to technology. Despite the existence of different factors motivating substitution, the *replacement* and not *enrichment* (along the lines of additive and subtractive bilingualism [21]) of technology use can point at structural issues. Grin [20] created a framework of Capacity, Opportunity, and Desire to capture the factors at play in language vitality [7]. A community's desire to use a language in digital spaces should always be matched by appropriate opportunities to do so. Consequently, technology development needs to be coordinated with communities – a mismatch between the desired uses and available opportunities can start the vicious circle of substitution to the detriment of excluded communities.

This issue exhibits features of the Matthew Effect [23], whereby well-resourced languages receive increasingly more and better applications and technological solutions for NLP tasks, while the rest is cast further adrift – in other words, the digital divide is widening. This brings considerations about our research and development of applications into the political sphere, and forces us to give thought to the social implications of our research activities. The decisions we make directly affect the system and may aggravate the negative feedback loop: some low-resource languages have less support for data collection or curation, requiring more time and effort to have data sets conform to steadily rising requirements. As a result, each speaker or researcher associated with the creation, annotation, or curation of language data for these languages has to bear a higher individual burden. The reaction of researchers and community members to this adverse situation can provide us with insights for further developing this field.

3 Reactions and breaking the vicious circle

There are different frameworks which could be used to discuss the reactions shown by community members but, irrespective of a particular context, we can find that the existing structures [19] lead many users to opting for applications in languages other than their preferred one, whether it is for quality (e.g., accuracy of translations), coverage (e.g., specialised vocabulary), or ease of use and accessibility of the other application. This practice of substitution reinforces the adverse structures and facilitates the expression of the relative status or symbolic power of the majority community [9]. Some may show reactance [10] and support the creation of NLP applications for their languoid but, as discussed above, they do so at a high individual burden. We certainly cannot blame those users who opt for more developed resources and applications which cover their needs better, but we must consider the structures which reproduce and reinforce these inequalities and do not provide opportunities for using minority languages. One aspect already addressed critically in the previous section is standardisation and the benchmarking of the best. While it must be our goal to improve our standards and set high targets, we must be aware of the exclusive nature of these reference points which not all languages can meet. By labelling an application "state of the art", we make it desirable and prestigious. At the same time, the combination of high standards/requirements, size of the community, and the status of a language form influence the set of languoids which can successfully aspire to this prestigious technological resources, while the rest suffers from technological exclusion. The vicious circle continues through our individual actions and reactions to the structures – lowering standards, increasing community sizes and user groups, or enhancing the status of a language form are possible solutions but these, generally, lie beyond the remit of the individual researcher.

Looking at the scientific process involving language data in 2004, Nathan and Austin warn that a "consumerist approach" [25] will not support endangered, minoritised, or low-resource languages. There are two important points in that assessment: first, as the authors discuss, there is a metadata gap between collections of language resources in (documentary) linguistics and their subsequent uses in computational linguistics and NLP applications. This disenfranchises, first and foremost, the speakers and consultants who give us permission to record their language use, their stories, and use it for scientific discovery. Certainly, "giving back in ways that are meaningful and valuable to the communities" [6, pp.49–50] is necessary for acknowledging a reciprocal relationship between researchers and the consultants. At the same time, this does not justify extraction and mechanistic decontextualisation (for a criticism of terms like "mining" or "harvesting" in contexts of research on minority languages see

Davis' 2017 article [13]). This extraction does not just affect the original "producers" of data (i.e., speakers and consultants), but also all researchers, annotators, translators who support the creation of language resources and high-quality data sets. I have myself experienced this extraction of data I curated, which I was surprised to see copied on Wikipedia without proper citation or attribution. While this was remedied without any difficulties, it is indicative of the second aspect of the consumerist rhetoric: the omnipresence of data. In our everyday lives, especially in academia, we face large amounts of data – in some instances, we can freely decide which data sets we want to use in our work. Therein lies the problem of the consumerist approach, as most researchers mirror the community members' behaviour outlined above by preferring higher quality, standardised, easily accessible data sets with a large coverage. At the same time, those academics who show reactance and work on low-resource languages face a struggle against evolving standards and expectations which are benchmarked against well-resourced languages. Do we, as academics, also enter an vicious circle – is there a point where we stop caring for minoritised languages and their communities?

I would like to argue with the same frameworks of agency introduced above that we do possess agency as scientists [33]. Especially in instances where our decisions affect communities outside of academia, it is our responsibility to acknowledge this relationship through our decisions. This is relevant in our work with communities who contribute to our research by providing us with data, which we must respect and honour – not just as the documentary linguist who "collects" data (to use the extraction metaphor) but also as the "consumers" of these data who have to acknowledge not only the speakers and consultants but also their colleagues in linguistics or anthropology who enriched data sets and made them usable. All of these stages of "language work" [22], as part of the scientific process, should aim for an appreciative stance of data and its producers. In terms of data citation, there are positive developments towards this goal, e.g., the Tromsø recommendations [4], although I would contend that we must extend that to past publications [30] with a view to preserving the human traces in our work, as also argued by Nathan & Austin [25] and termed "finding the human in the loop" by Bird [8]. Reaching this stance of acknowledging all contributions to a data set can furthermore help to prevent biases [32], as we can, ideally, track and reconstruct links through time and different layers of data [31, discussed under the notion of "Metadata Inheritance"]. These steps are possible for every researcher working with language data and conducting "language work", and leads to a more reciprocal relationship between communities and different groups of scientists.

4 Outlook

The goal of this paper is to highlight different gaps which are widening, as we have been able to observe for the past decade, and which we should aim to close through mindful decisions in our research. These gaps do not only exist between different language communities as technological exclusion – created through different language status, population size, or differences between requirements and capacities – but also between groups of researchers. If the consumerist approach prevails, colleagues working in linguistics departments will be relegated to producers of language data in competition with each other (if we continue with the economicist notion of academia).

This is not a one-way process where consultants provide linguists with data who produce resources and data sets, from which the computer scientist can pick at will and without bearing responsibility. Those who have accepted this responsibility and support communities and colleagues creating data sets deserve utmost respect (e.g., members of the ACL SIGEL,

the ComputEL community, or other special interest groups focusing on minority languages). Yet, to overcome the technological exclusion and support low-resource languages, more scholars need to critically assess how their decision-making and pursuit of ever-increasing targets and standards impacts those suffering from structures of exclusion and extraction. If more researchers subscribed to these goals, academia, in general, would be better equipped to support minoritised language communities by facilitating their participation (text processing, keyboards, dictionaries, spell-checking etc.). The precise goals have to be set by the communities and tailored to their needs, respecting their agency [8]; scholars and their institutions have access to the knowledge, the tools, and the funding to realise these projects. In turn, the technological support to minority languages can initiate a virtuous circle, whereby communities can catch up to the standards for more advanced, intelligent, or high-quality NLP applications, if this is seen as a goal for the community members. Considering ways of closing the gap will halt exclusion based on size and status of languages and may set a signal for the community by increasing the prestige of their language. Yet, for this to happen, we need to assess our goals as scholarly community and also address structural problems that encourage and reward those who take the easy path of where the ready-made data sets are. Not only are these options at times easier, research on major languages still attracts more funding, creates opportunities for fast production and output of publications, and tempts researchers with prizes, awards, and reputation. The goal is not to blame and shame colleagues who go down this route or to discredit their work – but to be aware that languages and data are not commodities that can be consumed or exchanged at will, without having implications on language communities. Handling language data and developing NLP applications always brings ethical considerations about social and political consequences of this work, and neither the community of linguists nor the community of computer scientists can escape their responsibility.

Linguistic justice, equal chances for participation, and the acknowledgement of the producers of language data through "giving back" are strong ethical arguments for the stance outlined above. Yet, we can further emphasise the benefits of devoting time to low-resource language to computer scientists. First, some may embrace the multiple challenges which low-resource languages pose and see these contexts as a chance to test new approaches [28]. Second, increasing the amount and quality of language data sets enables testing different approaches across a variety of data, allowing for testing hypotheses about the applicability and quality of applications. For linguistics, it has been argued that minority languages help in testing assumptions and theories by creating diverse data sets [29] and representing some "exceptional types" [26, p.367]. Third, some researchers have argued that minority languages are more likely to have retained rare linguistic features [34], which may not be stable in other languages [12]. Investigating these features can help linguists working on typology, while, simultaneously, allowing NLP scholars to test their applications and tools on languages outside of the commonly used national languages of Indo-European or Standard Average European typology. As a result, the improved and tested applications can be used to create even more data sets, the reversal of the vicious circle. Students and early career researchers should be made aware of these opportunities, while senior researchers, supervisors, and funding bodies can help with creating a conducive environment for starting a virtuous circle.

The perspective presented in this paper is strongly advocating the case of minoritised communities and low-resource languages, and is unlikely to be adopted by all readers. There are numerous arguments in favour of advancing technological standards and computational methods for major languages, which also support minority communities. But the challenge for the next decade is to foster awareness of ways in which our decisions as researchers serve

to reproduce and reinforce inequalities and technological exclusion. With 2022–2032 being declared the Decade of Indigenous Languages by the United Nations, the upcoming years are the best time to embrace the needs of indigenous communities and minorities as our research priorities. The first step in overcoming the consumerist approach and considering reciprocity consists in adopting a critical view on the research we conduct and present. A presentation or research outline which – without reflection – follows the rhetorics of "*I downloaded the gold standard corpus with X million tokens*" will not do justice to under-resourced or minoritised languages. These data sets will generally have been created and curated by speakers and linguists, and do not simply exist like products on a goods shelf. Instead, we should consider ways of making our research applicable and usable for other researchers and practitioners doing language work, thereby making it sustainable, as well as supporting communities who suffer from technological exclusion.

References

1. Oliver Adams, Adam Makarucha, Graham Neubig, Steven Bird, and Trevor Cohn. Cross-lingual word embeddings for low-resource language modeling. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 937–947, Valencia, Spain, 2017. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/E17-1088.
2. Željko Agić, Dirk Hovy, and Anders Søgaard. If all you have is a bit of the Bible: Learning POS taggers for truly low-resource languages. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 268–272, Beijing, China, 2015. Association for Computational Linguistics. doi:10.3115/v1/P15-2044.
3. Željko Agić, Anders Johannsen, Barbara Plank, Héctor Martínez Alonso, Natalie Schluter, and Anders Søgaard. Multilingual Projection for Parsing Truly Low-Resource Languages. *Transactions of the Association for Computational Linguistics*, 4:301–312, 2016. doi:10.1162/tacl_a_00100.
4. Helene N. Andreassen, Andrea L. Berez-Kroeker, Lauren Collister, Philipp Conzett, Christopher Cox, Koenraad De Smedt, Bradley McDonnell, and the Research Data Alliance Linguistic Data Interest Group. Tromsø recommendations for citation of research data in linguistics, 2019. doi:10.15497/rda00040.
5. Mikel Artetxe and Holger Schwenk. Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610, 2019. doi:10.1162/tacl_a_00288.
6. Peter K. Austin. Communities, ethics and rights in language documentation. In Peter K. Austin, editor, *Language Documentation and Description*, volume 7, pages 34–54. SOAS, London, 2010.
7. Joseph Lo Bianco and Joy Kreeft Peyton. Vitality of heritage languages in the united states: The role of capacity, opportunity, and desire. *Heritage Language Journal*, 10(3):i–viii, 2013. doi:10.46538/hlj.10.3.1.
8. Steven Bird. Decolonising speech and language technology. In Donia Scott, Nuria Bel, and Chengqing Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3504–3519, Barcelona, 2020. International Committee on Computational Linguistics. doi:10.18653/v1/2020.coling-main.313.
9. Pierre Bourdieu. *Language & Symbolic Power*. Polity Press, Malden, 1991.
10. Jack Brehm. *A theory of psychological reactance*. Academic Press, New York, 1966.
11. Michael Cysouw and Jeff Good. Languoid, doculect and glossonym: Formalizing the notion "language". *Language Documentation & Conservation*, 7:331–359, 2013.

12 Michael Cysouw and Jan Wohlgemuth. The other end of universals: theory and typology of rara. In Jan Wohlgemuth and Michael Cysouw, editors, *Rethinking Universals*, pages 1–10. De Gruyter Mouton, 2010. doi:doi:10.1515/9783110220933.1.

13 Jenny L. Davis. Resisting rhetorics of language endangerment: Reclamation through indigenous language survivance. In Wesley Y. Leonard and Haley De Korne, editors, *Language Documentation and Description*, volume 14, pages 37–58. EL Publishing, London, 2017.

14 Lise Dobrin, Peter K. Austin, and David Nathan. Dying to be counted: the commodification of endangered languages in documentary linguistics. In Peter K. Austin, editor, *Language Documentation and Description*, volume 6, pages 37–52. SOAS, London, 2009.

15 Long Duong, Trevor Cohn, Karin Verspoor, Steven Bird, and Paul Cook. What can we get from 1000 tokens? a case study of multilingual POS tagging for resource-poor languages. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 886–897, Doha, 2014. Association for Computational Linguistics. doi:10.3115/v1/D14-1096.

16 Meng Fang and Trevor Cohn. Model transfer for tagging low-resource languages using a bilingual dictionary. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 587–593, Vancouver, 2017. Association for Computational Linguistics. doi:10.18653/v1/P17-2093.

17 Candace Kaleimamoowahinekapu Galla and Alanaise Goodwill. Talking story with vital voices: Making knowledge with indigenous language. *Journal of Indigenous Wellbeing*, 2(3):67–75, 2017.

18 Duncan Gallie, Serge Paugam, and Sheila Jacobs. Unemployment, poverty and social isolation: Is there a vicious circle of social exclusion? *European Societies*, 5(1):1–32, 2003. doi:10.1080/1461669032000057668.

19 Anthony Giddens. *The Constitution of Society*. University of California Press, Berkeley, 1984.

20 François Grin. *Language Policy Evaluation and the European Charter for Regional or Minority Languages*. Palgrave Macmillan, Basingstoke, 2003.

21 Rodrigue Landry and Réal Allard. Beyond socially naive bilingual education : the effects of schooling and ethnolinguistic vitality on additive and subtractive bilingualism. *NABE Annual Conference Journal*, pages 1–30, 1993.

22 Wesley Y. Leonard. Producing language reclamation by decolonising 'language'. In Wesley Y. Leonard and Haley De Korne, editors, *Language Documentation and Description*, volume 14, pages 15–36. EL Publishing, London, 2017.

23 Robert K. Merton. The Matthew Effect in science. *Science*, 159(3810):56–63, 1968. doi:10.1126/science.159.3810.56.

24 Tom Moring. Functional completeness in minority language media. In Mike Cormack and Niamh Hourigan, editors, *Minority Language Media. Concepts, Critiques and Case Studies*, pages 17–33. Multilingual Matters, Clevedon and Buffalo and Toronto, 2007.

25 David Nathan and Peter K. Austin. Reconceiving metadata: language documentation through thick and thin. In Peter K. Austin, editor, *Language Documentation and Description*, volume 2, pages 179–188. SOAS, London, 2004.

26 Revere Perkins. The covariation of culture and grammar. In Michael Hammond, Edith Moravcsik, and Jessica Wirth, editors, *Studies in Syntactic Typology*, pages 359–378. Benjamins, Amsterdam and Philadelphia, 1988.

27 John E. Petrovic and Bedrettin Yazan. Language as instrument, resource, and maybe capital, but not commodity. In Bedrettin Yazan John E. Petrovic, editor, *The Commodification of Language*, pages 24–40. Routledge, London, 2021.

28 Edoardo Maria Ponti, Helen O'Horan, Yevgeni Berzak, Ivan Vulić, Roi Reichart, Thierry Poibeau, Ekaterina Shutova, and Anna Korhonen. Modeling language variation and universals: A survey on typological linguistics for natural language processing. *Computational Linguistics*, 45(3):559–601, 2019. doi:10.1162/coli_a_00357.

29 Jan Rijkhoff. Rara and grammatical theory. In Jan Wohlgemuth and Michael Cysouw, editors, *Rethinking Universals*, pages 223–240. De Gruyter Mouton, 2010. doi:doi:10.1515/9783110220933.223.

30 Tobias Weber. Can Computational Meta-Documentary Linguistics Provide for Accountability and Offer an Alternative to "Reproducibility" in Linguistics? In Maria Eskevich, Gerard de Melo, Christian Fäth, John P. McCrae, Paul Buitelaar, Christian Chiarcos, Bettina Klimek, and Milan Dojchinovski, editors, *2nd Conference on Language, Data and Knowledge (LDK 2019)*, volume 70 of *OpenAccess Series in Informatics (OASIcs)*, pages 26:1–26:8, Dagstuhl, Germany, 2019. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/OASIcs.LDK.2019.26.

31 Tobias Weber. Metadata Inheritance: New Research Paper, New Data, New Metadata? In Andrea Mannocci, editor, *Reframing Research Workshop Accepted Papers*. Zenodo, 2020. doi:10.5281/zenodo.4155362.

32 Tobias Weber. A philological perspective on meta-scientific knowledge graphs. In Ladjel Bellatreche, Mária Bieliková, Omar Boussaïd, Barbara Catania, Jérôme Darmont, Elena Demidova, Fabien Duchateau, Mark Hall, Tanja Merčun, Boris Novikov, Christos Papatheodorou, Thomas Risse, Oscar Romero, Lucile Sautot, Guilaine Talens, Robert Wrembel, and Maja Žumer, editors, *ADBIS, TPDL and EDA 2020 Common Workshops and Doctoral Consortium*, pages 226–233, Cham, 2020. Springer International Publishing. doi:10.1007/978-3-030-55814-7_19.

33 Tobias Weber and Mia Klee. Agency in scientific discourse. *Bulletin of the Transilvania University of Braşov Series IV: Philology and Cultural Studies*, 13(1):71–86, 2020. doi:10.31926/but.pcs.2020.62.13.1.5.

34 Jan Wohlgemuth. Language endangerment, community size and typological rarity. In Jan Wohlgemuth and Michael Cysouw, editors, *Rethinking Universals*, pages 255–278. De Gruyter Mouton, 2010. doi:doi:10.1515/9783110220933.255.

Representing the Under-Represented: a Dataset of Post-Colonial, and Migrant Writers

Marco Antonio Stranisci
Department of Computer Science, University of Turin, Italy

Viviana Patti
Department of Computer Science, University of Turin, Italy

Rossana Damiano
Department of Computer Science, University of Turin, Italy

Abstract

In today's media and in the Web of Data, non-Western people still suffer a lack of representation. In our work, we address this issue by presenting a pipeline for collecting and semantically encoding Wikipedia biographies of writers who are under-represented due to their non-Western origins, or their legal status in a country. The two main components of the ontology will be described, together with a framework for mapping textual biographies to their corresponding semantic representations. A description of the data set, and some examples of biographical texts conversion to the Ontology Classes, will be provided.

2012 ACM Subject Classification Information systems → Ontologies

Keywords and phrases Ontologies, Knowledge Graph, Language Resources, Migrations

Digital Object Identifier 10.4230/OASIcs.LDK.2021.7

Funding The work of Marco Antonio Stranisci and Viviana Patti is partially funded by the project "Be Positive!" (under the 2019 "Google.org Impact Challenge on Safety" call).

1 Introduction

Social media, and other User Generated Content platforms have given voice to an unprecedented number of people, while the Semantic Web offers encyclopedic knowledge about the world in an open, machine readable format. However, such technological transformation has not completely resulted in a more pluralist communicative environment, because the voices of people from non-Western countries are often unheard in crucial contexts. For instance, the involvement of minority journalists in mainstream newspapers is an open issue [23], as long as the integration of post-colonial perspectives within school textbooks [21]. This under-representation could be problematic since it precludes a full appreciation and understanding of diversity in our society.

The Under-Represented Writers (URW) project[1] aims at reducing this under-representation through a semantic modeling of authors whose biographies are characterized by belonging to a former colony country or being migrant. Its aim is twofold: encoding their lives in a non-stereotypical way, and providing a publicly available, semantically encoded knowledge source about them.

Modeling the biography of a writer who is potentially under-represented due to his ethnicity raises two main issues addressed in our work. (1) The interaction of the attributed ethnicity of a person, which is a subjective concept, with her/his legal status in the place where she/he lives. The project relies on an ontology to provide an explicit and objective

[1] The project is available at `https://w3id.org/UnderRepresentedWritersOntology/`

representation of this interplay, further specialized to describe the citizenship laws of a particular set of countries. (2) The knowledge extracted from Linked Data sources in the form of RDF triples does not allow arranging the legal statuses of a person in a coherent whole during her/his lifetime, since it relies on a set of vocabularies that have not been designed to express this type of knowledge. However, representing the biography of a writer in terms of the relations with different countries along time is crucial because it allows interpreting her/his literary production in the light of the social context in which she/he was situated when she/he created them.

The Under-Represented Writers (URW) ontology was used to gather a collection of writers, and their biographies in English language from Wikidata, and DBpedia. The resulting Knowledge Graph expresses an ordered and systematic list of features about authors' birthplace and time of birth, together with their legal statuses along time, all the facts about their lives gathered from Wikipedia, and a mapping of verbs in their biographies according to the ERE ontology [3]. Writers' countries of birth are classified along three dimension: their status of former colony, their Human Development Index score, and their mobility score.

The paper is structured as follows: in Section 2, a review of the related work is introduced. Section 3 presents the ontology: the formalization of the interplay between ethnicity, and legal status is described in 3.1, while the representation of life events is explained in 3.2. Section 4 provides a description of the data gathering process (4.1), together with an overview of the Knowledge Graph (4.2). In Conclusion (Section 5), results and open issues are discussed.

2 Background and Related Works

The project described here relies on three main lines of research. In Section 2.1, literary and narratives theories that guided the ontology design process are presented. Then, an overview of the related work on semantic representation of biographies (Section 2.2), and event annotation models (Section 2.3) is provided.

2.1 Post-Colonial Literatures, and Post-Classical Narrative Theories

Post-Classical narrative theories [24] were born during the Eighties as an alternative to the semiotic approach to narratives. Instead of focusing only on texts, scholars started investigating the economic and political contexts in which cultural works are produced, thus highlighting the strong interconnection between the author and the cultural norms and values shaping her/his narratives ([18, 4]). This paradigm shift led to a wide set of theories, such as feminist narratology [15] and ethnic narratology [8].

Alongside the spreading of post-classical approaches to the study of narratives, the heterogeneous field of research labeled under the term "Post-Colonial studies" raised interest around the issue of under-representation of former colony country citizens' voices. "In the context of colonial production, the subaltern has no history and cannot speak" [33] due to an epistemic violence perpetrated by European countries through a systematic practice of silencing [7]. Non-dominant indigenous groups did not take part to the elaboration of their countries' official culture and history, while local élites were raised with a Western education. Besides the usage of violence, Europeans enacted a textual takeover [10] of non-Western countries by imposing their cultural traditions. During this process, colony citizens suffered a linguistic and physical displacement [1], since they lost control on their countries, and languages. Post-colonial literature is a heterogeneous cultural project aimed at reaching emancipation from the colonizer countries.

A complementary problem affects the reception of migrants narratives within the context of European countries. Recent studies showed that the fortune of novel from ethnic minorities is often related to an ethnographic interest in exoticism, and not to a need of a deep understanding of other cultures [12]. Readers, rather than being interested in the literary work itself, focus on how it conveys information about immigrant writers' ethnic identities. In response to this expectation, some of them deliver stereotypical representation of their ethnicity in their novels [22].

Our project acknowledges the relevance of political and social factors in studying the narratives produced by writers by explicitly modeling in the ontology the conditions leading to a lack of representation. More to the point, our attempt is to describe three types of biographical situations potentially correlated with under-representation: living in a former colony country, being a migrant, and belonging to an ethnic minority.

2.2 Biographic Ontologies

Two projects have tackled the issue of collecting and describing the biographies of writers, and under-represented people. The CWRC Ontology[2] [2, 31] has been developed to support the Orlando project[3]: a data set of 1,300 women British writers aimed at widening the study of feminist literary research. The ontology has an extensive taxonomy of classes describing the biography of a woman writer with the set of characteristics that determines her condition, such as ethnicity, political affiliation, reproductive history, and sexuality.
The Enslaved Ontology[4] [30] is a modular ontology aimed at mapping several databases about African slavery in a single Knowledge Graph[5] aligned with Wikidata [38]. Similarly to the Orlando project, the Enslaved Ontology models socio-cultural information about people in the data set, in order to reconstruct the social networks characterizing slavery. However, controlled vocabularies were chosen – rather than formalised concepts – to express detailed information about persons and events.

Our Ontology shares some similarities with these projects, as long it aims at representing a group of persons sharing a specific condition, such as belonging to an ethnic minority. However, the concept of "being under-represented" is challenging from the modeling perspective, because it has blurred boundaries and it can be very subjective. Furthermore, our project intentionally does not model ethnographic features, choosing instead to fully describe the interplay between a person and the places where she/he lives during her/his life.

Some proposals are specifically targeted at the representation of biographies. The Biography Ontology [14, 13] models biographical events as time-dependent knowledge by directly adding temporal arguments to a materialised triple. In the example below, the marriage between Tony Blair and Cherie Booth is first expressed, then its temporal boundaries are added, together with other optional information. Semantic conciseness characterizes such approach, which has a major drawback in the generalization of complex events determined by multiple factors.

```
    tony blair marriedTo cherie booth
‘‘1980-03-29’’xsd:date ‘‘2015-05-08’’^^xsd:date
    location London
```

[2] http://sparql.cwrc.ca/ontologies/cwrc.html
[3] http://www.artsrn.ualberta.ca/orlando/
[4] https://docs.enslaved.org/ontology/
[5] https://enslaved.org/

BKOnto [36] is built upon the Time[6] and the StoryLine[7] ontologies. Token-reified biographical events are arranged in StoryLine slots, and further decomposed to express more detailed spatial and temporal information about them. The BIO vocabulary[8] collects 34 types of life events which can be used to create a biographical timeline.

Our work is aimed at modeling time-dependent knowledge like The Biography Ontology [14], but relies on the Ontology Design Pattern approach [27, 11], since it encodes the status of a person resulting from a combination of *roles* she/he experiences in a given situation. For such reason, a rich account of modular and expressive legal statuses related to citizenship and discriminatory factors are necessary, rather than a closed taxonomy of biographical events.

The study of prosopography as a methodological tool for historical research is the object of the Factoid Prosopography Ontology (FPO) [26]. Leveraged by several projects targeted at the study of Middle Ages in Europe and Asia[9], FPO connects the representation of personal factoids (such as birth, death, acquisition of goods or social status) with the documentation about them (e.g., legal statements, seals, and other artifacts), so as to support a systematic investigation of biographies through documents. In our project, the availability of authoritative sources, stored in digital form, about the identity and status of the biographies under study makes the representation of documentary sources less relevant than it would be when dealing with ancient ages, bringing instead into focus the unambiguous definition of domain-specific notions such as citizenship and migration.

2.3 Event Encoding

Many approaches aimed at encoding and annotating events have been proposed in the last years. Despite the common representational goal, they vary significantly, since events can be formalized at different levels of granularity.

The Clinical Narrative Temporal Relation Ontology (CNTRO) [35] provides a representation of clinical events offering a comprehensive taxonomy of temporal representations mapped onto authoritative representations of time [34].

The Story Intention Graphs (SIG) [9] encodes the intentions of narrative agents by linking them to textual fragments. Its application to personal narratives has been proposed by [17].

An approach sharing a similar granularity is the one proposed by [5]. Here, events are frame-like structures of the type <Agent, Predicate, Theme, PP> whose affective polarity is annotated.

Finally, there are several works that analyze events at a word level. The ACE/ERE projects [6, 32] rely on the identification of the events through the annotation of "Trigger" words or multi-word expression. The TimeML annotation scheme [28] has been specifically designed for identifying temporal expressions in a text, and annotating the chronological relation between them. Finally, the Richer Event Description (RED) framework [25] simplifies the taxonomy of events proposed in TimeML, but adds information about the causal relationships over them.

Even if all these approaches contribute to an exhaustive representation of events in texts, a unifiying model that systematically links the syntactic and the semantic layers of an event is still missing. Our work tries to bridge together these two levels by mapping ERE's "Trigger" verbs of movement and Wordnet synsets, according to Semantic Web principles[19].

[6] https://www.w3.org/TR/owl-time/
[7] https://www.bbc.co.uk/ontologies/storyline
[8] https://vocab.org/bio/
[9] https://www.kcl.ac.uk/factoid-prosopography/projects

3 The Ontology of Under-Represented Writers

The Ontology of Under-represented Writers (URW) is an attempt to provide a formal and objective description of authors who potentially suffer a lack of prominence due to the context where they were born. The URW is aimed at highlighting the biographical events, and situations during which a writer may has been experienced the condition of being a subaltern [33].

With these goals in mind, we designed the URW Ontology to answer the following Competency Questions:

- What people, born in a former colony country, wrote at least one literary work while living in their birthplace?
- Which are the writers who experienced the condition of being migrant?
- Which second generation migrants or minorities wrote at least one literary work?

In the next sections, a description of how our semantic model fits with the first two Competency Questions is provided. The encoding of second generation migrants, and minorities and is not addressed in this work, due to the lack of information about ethnicity, rarely specified in Wikidata (Section 4).

3.1 Modeling the Interplay between Ethnicity and Legal Status of a Person

Since the post-colonial framework is irreducible to a unifying taxonomy (see Section 2.1), it is crucial to describe the concept of under-representation related to ethnicity without falling into arbitrary categorizations. Hence, we propose an agnostic model that operates through the intersection of two elements: the country where a person was born, and information about her/his family relationships.

The country of birth is the first feature to express under-representation. Instead of citizenship and ethnicity, which may change along time (the former), and be subjective (the latter), the fact of being born in a given place is an immutable, closer to objectiveness property. So, concerning the country of birth, we defined three indicators that may correlate with the under-representation of a writer:

- its status as a former colony;
- the mobility score of its passport[10];
- its Human Development Index[11], aimed at excluding rich former colonies from the collection, such as Singapore or Israel.

Finally, family relationships are used to determine whether a person is a second generation migrant or if she/he belongs to an ethnic minority in a given country. The interplay of these two features (country and family) helps to determine the condition of under-representation. However, this model can be adapted to other domains of knowledge in which a representation of the relation of a person, her/his birth country, and her/his family network is needed. In Figure 1, a graphical representation of Chimamanda Ngozi Adichie (a contemporary Nigerian writer) birth country is provided. In this case, the simultaneous membership of Nigeria to sets of different classes (urw:CountryWithLowHDI, urw:CountryWithLowMobilityScore, and urw:FormerColonyCountry) is a signal of a *potential* lack of representation of this author.

[10] https://www.passportindex.org/
[11] http://hdr.undp.org/en/content/human-development-index-hdi

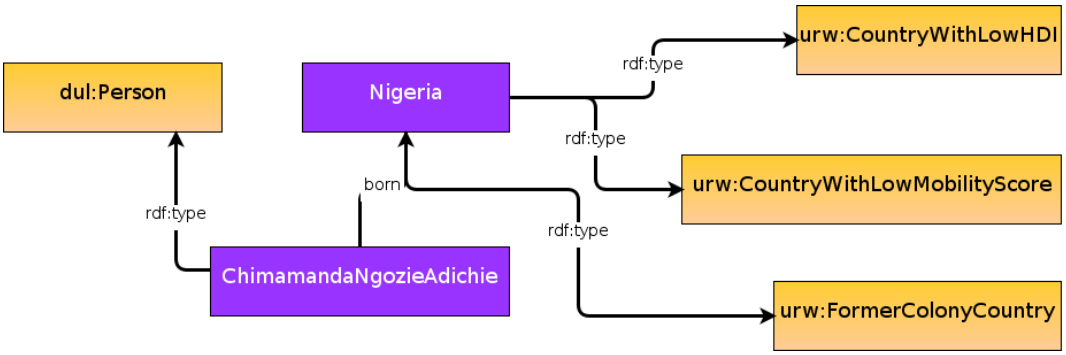


Figure 1 The representation of Chimamanda Ngozie Adichie place of birth, that lead to consider her an under-represented writer.
Within the paper, classes are represented with yellow boxes, while purple boxes identify individuals.

3.2 Modeling Biographies

As mentioned in Section 2.2, there are some proposed approaches for encoding a biography. For our purpose of modeling the life of an under-represented person, two kinds of situations need to be described: the process of migrating, and the status of a person in a given country. Both are embodied in a specific time interval, and this relation of time-dependency need to be formally expressed for two reasons: on one side, it is essential to order life events in a chronological fashion; on the other side, it allows drawing a link between a writer's life, and her/his cultural production. Our solution relies on the Ontology Design Pattern (ODP) framework, since it provides foundationally sound, re-usable building blocks for representing common patterns across ontologies, with advantages for design and interoperability. More specifically, we adopted the BasicPlanExecution ODP to describe a migration, since a migration represents the execution of a intentionally devised line of action, and the TimeIndexedPersonRole for modeling the legal status of a person, because the legal status of a person with respect to a country is typically non-rigid and can be modelled as a role.

The URW:MIGRATION class (see Figure 2) is subclass of the DUL:PLANEXECUTION class and, as such, **dul:isSettingFor** six elements: the action of URW:MIGRATING, which is an event, a DUL:PERSON, namely the agent who is migrating, and her/his URW:MIGRATIONROLE in the migration process. The URW:MIGRATION class is also a setting for the spatio-temporal coordinates of the migration: the DUL:TIMEINTERVAL along which it occurs, the URW:PLACEOFARRIVAL of the migration, and its URW:PLACEOFDEPARTURE. We modeled the reasons for a person to leave her/his country for another as a URW:MIGRATIONREASON that is part of a URW:MIGRATIONPLAN satisfied by the URW:MIGRATION situation. The URW:MIGRATIONREASON **dul:defines** the URW:MIGRATIONROLE of a person.

The URW:TIMEINDEXEDPERSONSTATUS class **dul:isSettingFor** a DUL:PERSON and her/his CONDITIONROLE. Furthermore, it **dul:hasComponent** a DUL:CLASSIFICATION, which **dul:isSettingFor** a DUL:TIMEINTERVAL, a DUL:PLACE, and a URW:CONDITION. The latter is primarily used to express the legal status of a person (eg: citizen, economic migrant, refugee), but it also can be used to describe other features that determines her/his condition. For instance, religion, sexual orientation, or social class. These additional aspects currently fall outside the scope of the ontology, so they are purposely left open to the integration with other semantic resources, such as ontologies (e.g. [2]) or controlled vocabularies (e.g. [30]).

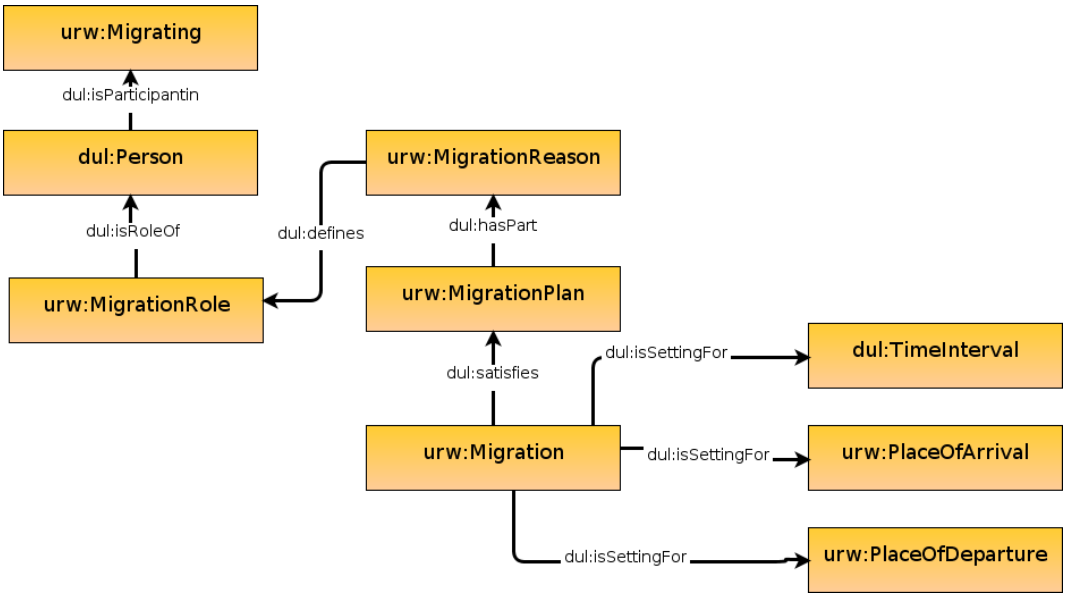


Figure 2 A graphical representation of the URW:MIGRATION class. The "urw" prefix stands for Under-Represented Writer Ontology; "dul" is the prefix of Dolce, the upper ontology which is the reference for foundational concepts in our work.

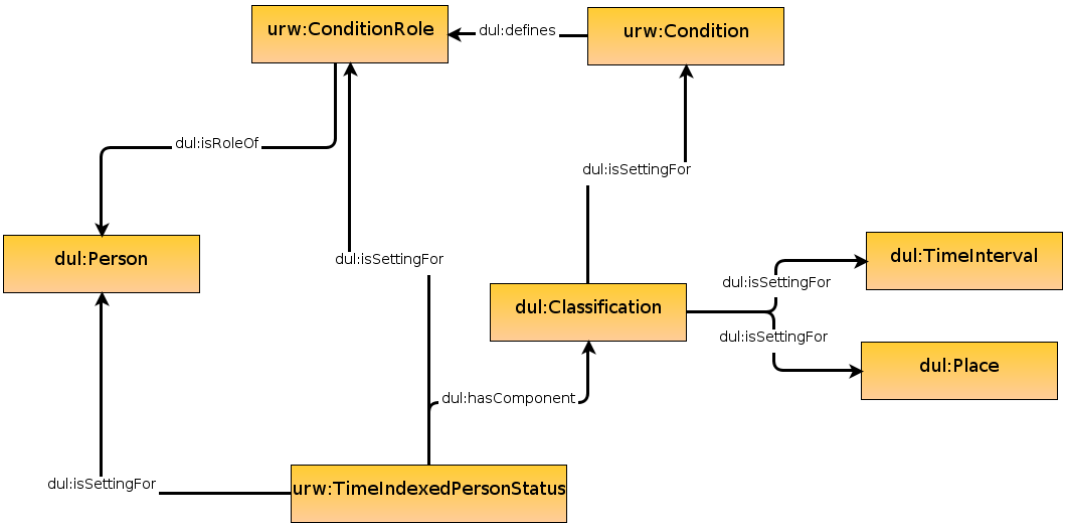


Figure 3 A graphical representation of the urw:TimeIndexedPersonStatus class.

3.3 Integration with Other Ontologies

In addition to the uw:TimeIndexedPersonStatus, and the uw:Migration, we partly integrated in the URW Ontology three other ontological resources. The above mentioned Named Authority List of countries maintained by the European Union[12]: an authoritative, comprehensive, and multilingual reference for country names. In our project, its main use is to standardize information about authors biographical places.
The PROV-O [16] Ontology is a standard to express the provenance information of a work. In this context, its use has two aims: making explicit the sources of information about writers biographies; pairing authors to their works.
The Ontolex-Lemon model [20] semantically enriches the event triggers defined within the ERE project [3], by mapping the morphological and syntactic properties of lexical entries to the semantic categories expresses by Classes. This is supposed to facilitate the process of converting the raw text of the authors' biographies to RDF triples, as described in Section 4.1.

4 The URW Knowledge Graph

In this section, we describe the first version of the URW Knowledge Graph, a collection of writers and biographies from Wikidata. After describing the data gathering process, we give an overview of the data set and provide some examples of the encoded entities.

4.1 Data Gathering

A preliminary identification of features about authors to be included in the Knowledge Graph led us to disregard ethnicity. Such information can introduce a bias in the collection because of the demographics of Wikipedia editors [37]. Moreover, the ethnic group of an author is available only in the 4.8% of the cases. The birthplace and time of birth appeared to be two widespread and objective features, instead. Therefore, data gathering has been devoted to obtain this information for each author. The data collection pipeline (see Figure 4) consisted of several steps:

- First, we collected through the Wikidata Query Service[13] all the instances (corresponding to the class WDT:P31 in Wikidata) of type human (WD:Q5), which has occupation (WDT:P106) of type novelist (WD:Q6625963), poet (WD:Q49757), or writer (WD:Q36180). We thus obtained 246,574 records.
- Then, we obtained the dates of birth (WDT:P569), by using the Pywikibot Python library[14]. In order to avoid duplicates or data misalignment, we only stored the year of birth of each writer, of available. The collection was reduced to 227,840 items.
- We then filtered only writers from a specific historical period to nowadays. We chose the Berlin Conference held in 1884, which formally started the scramble for Africa, an emblematic moment of the European textual takeover of non-Western world, directly related to the subsequent Decolonization and spreading of post-colonial literatures. 155,294 authors were born from 1884.
- Next, for each collected author, we queried her/his place of birth (WDT:P19). The information about birth place is very heterogeneous in Wikidata: it can be a country,

[12] `https://op.europa.eu/en/web/eu-vocabularies/dataset/-/resource?uri=http://publications.europa.eu/\resource/dataset/country`
[13] `https://query.wikidata.org/`
[14] The library, available at `https://github.com/wikimedia/pywikibot`, was also used to collect places of birth and their corresponding countries.

Figure 4 The diagram representing the data gathering pipeline.

an administrative region, a city or even a district. In order to align all the birthplaces to a common format, we further queried the countries (WDT:P17) of all birthplaces. Throughout all this process, we used Europeana Eurovoc as an authoritative source to align all the geographical entities.

- Finally, we associated the Human Developed Index, the mobility score, and the eventual status of former colony to each country in order to group writers according to their level of under-representation. The resulting Knowledge Graph includes 127,141 authors.

4.2 KG Description

For each urw:Author of the Knowledge Graph the urw:CountryOfBirth, and urw:YearOfBirth are specified. Moreover, the urw:wikipediaText data property contains the reference text, expressed as a string. At a first glance (see Table 1), the URW Knowledge Graph shows an unbalanced distribution of writers across different continents[15]. More specifically, the data set seems to be Eurocentric, since the 64% of the authors were born in this continent. On the opposite side, African writers are the less represented amount of population taken into account. Finally, the number of individuals is dramatically reduced to 45,793 if we consider only the ones with a Wikipedia page in English. This drop mainly affects Europe and Latin America continents.

Table 1 The list of writers stored in the URW Knowledge Graph, divided by continent of birth.

Continent	Population	Writers on Wikidata	People per writer	Authors with English page
Africa	1,340,598.113	3,528	374,259.6	53.6%
Asia	4,641,054.786	14,993	309,548.1	45.9%
Europe	747,636.045	81,832	9,136.2	22.6%
Latin America	653,962.332	9,643	67,817.3	28.2%
North America	368,869.644	17,389	21,212.8	77.5%
Oceania	42,677.809	1,365	31,265.7	85.9%

4.3 Event Encoding Description and Examples

Wikipedia authors pages are concise texts providing a limited taxonomy of biographical facts: educational background, personal life events, movements, works, and awards. Such a homogeneous narrative style facilitates the extraction and encoding of migration-related

[15] Both the 6-continents model and the estimation of population by continent were taken from the UN Department of Economic and Social Affairs: `https://population.un.org/wpp/`

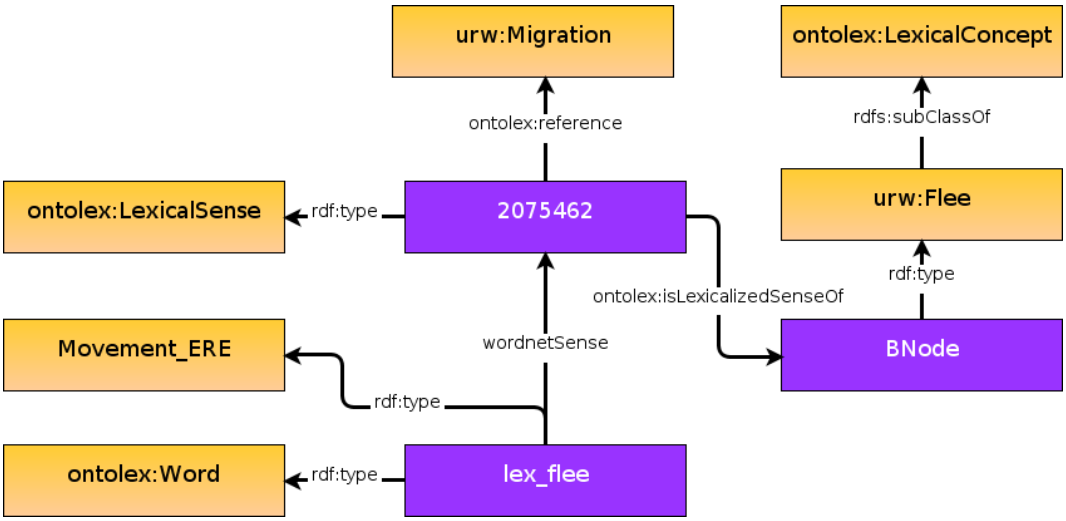


Figure 5 The encoding of the Movement_ERE event trigger *flee* mapped to the corresponding Wordnet offset. The prefix "ontolex" refers to the OntolexLemon ontology adopted for the mapping.

patterns (see 3.2) from raw text using preexisting linguistic resources: REO Ontology [3], and Ontolex-Lemon [19, 20]. Our conversion process encompassed two passages: (1) The collection of all the "Trigger" verb of "Movement_ERE" events in the REO Ontology [3], and the identification their occurrences in each biography. (2) The mapping of Movement_ERE "Triggers" with Wordnet offsets in RDF triples according to the Lemon methodology [19, 20].

According to the Ontolex-Lemon specification, each ONTOLEX:LEXICALENTRY (a word, a multiword expression of an affix) has a corresponding set of ONTOLEX:LEXICALSENSE. In our case, every ONTOLEX:LEXICALSENSE is a Wordnet offset, namely the lexicalized sense of an ONTOLEX:CONCEPT. The latter represents the mental concept evoked by a lexical entry. Finally, the ONTOLEX:LEXICALSENSE has a semantic reference within the Ontology, in our case a URW:MIGRATION or a URW:TIMEINDEXEDPERSONSTATUS.

Figure 5 is a graphical representation of how the verb "to flee" is encoded in the ontology. The ontolex:lex_flee is both a ONTOLEX:WORD and an EREONTOLOGY:MOVEMENT_ERE "Trigger", namely a textual token expressing an event in which a person moves from a place to another. The ontolex:lex_flee has a ONTOLEX:LEXICALSENSE, which in our case is the Wordnet offset id number of the verb flee (2075462), related to the definition "run away quickly". The Wordnet offset is the lexicalized sense of the URW:FLEE concept evoked by the ontolex:lex_flee. Finally, the ONTOLEX:LEXICALSENSE has a **ontolex:reference** to the URW:MIGRATION class of the ontology (Section 3.2).

We provide two examples of manually encoded biographical events to illustrate the pipeline. The first, depicted in Figure 6, shows a conversion from raw text to urw:Migration Class (Section 3.2). Below, there is an excerpt of the Wikipedia page of Kim Thúy, a Vietnamese-born Canadian writer.

> At the age of ten, Thúy **left** Vietnam with her parents and two brothers, joining more than one million Vietnamese boat people **fleeing** the country's communist regime after the fall of Saigon in 1975. The Thúys **arrived** at a refugee camp in Malaysia, run by the United Nations High Commission for Refugees, where they spent four months.[16]

[16] `https://en.wikipedia.org/wiki/Kim_Thúy`

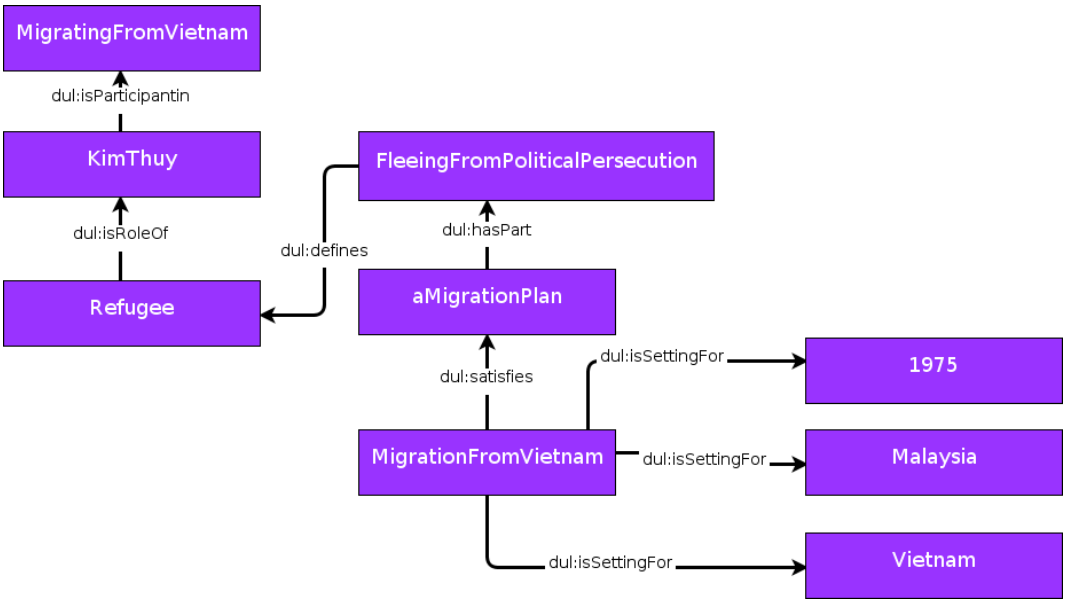


Figure 6 The encoding of a Kim Thúy childhood event as a uw:Migration.

Verbs in bold – *left*, *fleeing* (Figure 5), *arrived* – are "Triggers" of a movement event and identify a URW:MIGRATION situation and a URW:MIGRATING event, respectively labeled as urw:MigrationFromVietnam and urw:MigratingFromVietnam[17]. Vietnam, the URW: PLACEOFDEPARTURE, and Malaysia, the URW:PLACEOFARRIVAL, have been easily identified, since they are explicitly mentioned in the text. The individuation of the time when the URW:MIGRATION occurred was more difficult because a comparison between the expression "At the age of ten", and the URW:BIRTHYEAR of the writer was necessary to derive it by difference. Finally, textual references to "the fall of Saigon", and to "refugee camp" allowed identifying the URW:MIGRATIONREASON as fleeing from political persecution, and the URW:MIGRATIONROLE as a urw:Refugee.
It is worth mentioning that the occurrence of "run" in the example does not imply a movement. A more rigorous approach such as Lexico-Semantic Pattern [29] is needed to avoid false positives within an automatic conversion process.

The second example illustrates the text-to-urw conversion applied to Chimamanda Ngozi Adichie's biography (Figure 7). Again, the verb (*left*) is a trigger for the event of leaving a country for another, which allows identifying a URW:TIMEINDEXEDPERSONSTATUS instance, and the United States as the DUL:PLACE where the situation is experienced by the person. Similarly to the previous example, the beginning of the DUL:TIMEINTERVAL had to be inferred from the expression "At the age of 19". Several textual references to the concept of studying led to the individuation of the URW:CONDITION of being a urw:MigrantStudent. Finally, there are not any unambiguous verbal signals triggering the end of the experience of Adichie as a migrant student in the United States, as long as the reference to Yale University does not directly express the country which is the URW:PLACE of the URW:TIMEINDEXEDPERSONSTATUS. Again, a more sophisticated approach is needed, as it is pointed out in Section 5.

[17] This labeling was adopted to make the description clearer, since, in the Knowledge Graph, they are blank nodes

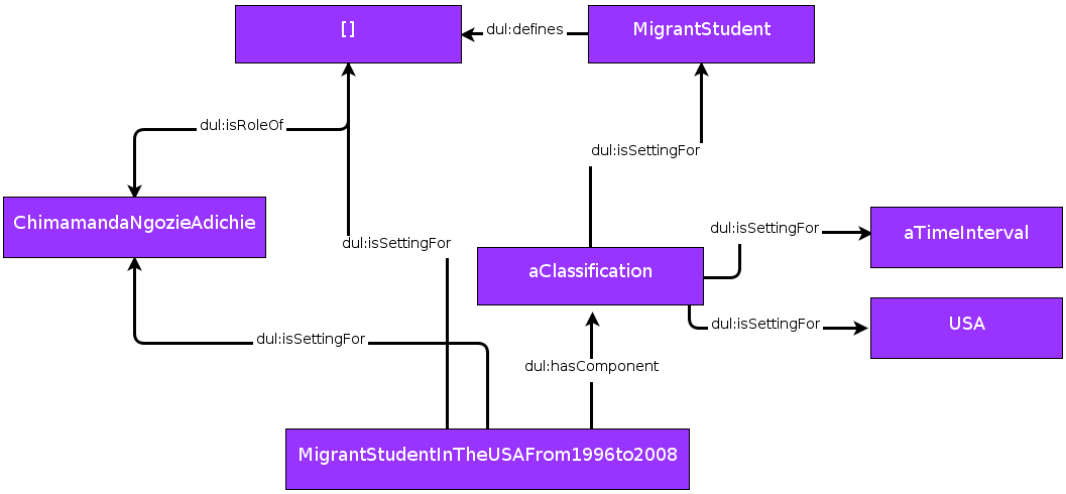


Figure 7 The encoding of the Adichie experience of migrant student in the United States according to the uw:TimeIndexedPersonStatus pattern.

> At the age of 19, Adichie **left** Nigeria for the United States to study communications and political science at Drexel University in Philadelphia.[...] In 2008, she received a Master of Arts degree in African studies from Yale University.[18]

5 Conclusion and Future Work

In this paper, we described the ongoing construction of a data set that relies on the URW ontology, a semantic model designed to encode the lives of migrant, post-colonial writers. After describing the ontology and the pipeline, we provided some examples of conversion from raw text biographies to URW:MIGRATION, and URW:TIMEINDEXEDPERSONSTATUS through the Ontolex-Lemon model. However, a systematic encoding has not been performed yet. This is a necessary step to gather, organize, and analyze narratives belonging to under-represented authors.

A first overview of the obtained Knowledge Graph shows that a lack of representation of non-Western authors is also present on Wikidata, together with the need to adopt a multilingual approach, since only the 36% of writers has an English Wikipedia page.

References

1 Bill Ashcroft, Gareth Griffiths, and Helen Tiffin. *The empire writes back: Theory and practice in post-colonial literatures*. Routledge, 2003.

2 Susan Brown, Patricia Clements, Isobel Grundy, Sharon Balazs, and Jeffrey Antoniuk. An introduction to the orlando project. *Tulsa Studies in Women's Literature*, 26(1):127–134, 2007.

3 Susan Windisch Brown, Claire Bonial, Leo Obrst, and Martha Palmer. The rich event ontology. In *Proceedings of the Events and Stories in the News Workshop*, pages 87–97, 2017.

4 L. E. Bruni. Cultural narrative identities and the entanglement of value systems. In *Differences, Similarities and Meanings: The Interplay of Differences and Similarities in Communication and Semiotics*. De Gruyter Mouton, In press.

[18] `https://en.wikipedia.org/wiki/Chimamanda_Ngozi_Adichie`

5 Haibo Ding, Tianyu Jiang, and Ellen Riloff. Why is an event affective? classifying affective events based on human needs. In *AAAI Workshops*, pages 8–15, 2018.

6 George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, and Ralph Weischedel. The automatic content extraction (ACE) program – tasks, data, and evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal, 2004. European Language Resources Association (ELRA). URL: `http://www.lrec-conf.org/proceedings/lrec2004/pdf/5.pdf`.

7 Kristie Dotson. Tracking epistemic violence, tracking practices of silencing. *Hypatia*, 26(2):236–257, 2011.

8 Laura Doyle and Laura Anne Doyle. *Bordering on the body: The racial matrix of modern fiction and culture*. Oxford University Press on Demand, 1994.

9 David K Elson. Detecting story analogies from annotations of time, action and agency. In *Proceedings of the LREC 2012 Workshop on Computational Models of Narrative, Istanbul, Turkey*, pages 91–99, 2012.

10 Leela Gandhi. *Postcolonial theory: A critical introduction*. Columbia University Press, 2019.

11 Aldo Gangemi and Valentina Presutti. Ontology design patterns. In *Handbook on ontologies*, pages 221–243. Springer, 2009.

12 Graham Huggan. *The postcolonial exotic: Marketing the margins*. Routledge, 2002.

13 Hans-Ulrich Krieger and Thierry Declerck. Tmo – the federated ontology of the trendminer project. In *LREC*, pages 4164–4171. Citeseer, 2014.

14 Hans-Ulrich Krieger and Thierry Declerck. An owl ontology for biographical knowledge. representing time-dependent factual knowledge. In *BD*, pages 101–110, 2015.

15 Susan S Lanser. Toward a feminist narratology. *Style*, pages 341–363, 1986.

16 Timothy Lebo, Satya Sahoo, Deborah McGuinness, Khalid Belhajjame, James Cheney, David Corsar, Daniel Garijo, Stian Soiland-Reyes, Stephan Zednik, and Jun Zhao. Prov-o: The prov ontology. Technical report, World Wide Web Consortium, 2013. URL: `https://www.w3.org/TR/prov-o/`.

17 Stephanie M Lukin, Kevin Bowden, Casey Barackman, and Marilyn A Walker. Personabank: A corpus of personal narratives and their story intention graphs. *arXiv preprint*, 2017. `arXiv:1708.09082`.

18 Dan P McAdams. Narrative identity. In *Handbook of identity theory and research*, pages 99–115. Springer, 2011.

19 John McCrae, Elena Montiel-Ponsoda, and Philipp Cimiano. Integrating wordnet and wiktionary with lemon. In *Linked Data in Linguistics*, pages 25–34. Springer, 2012.

20 John P McCrae, Julia Bosque-Gil, Jorge Gracia, Paul Buitelaar, and Philipp Cimiano. The ontolex-lemon model: development and applications. In *Proceedings of eLex 2017 conference*, pages 19–21, 2017.

21 Pia Mikander et al. Westerners and others in finnish school textbooks. *University of Helsinki, Institute of Behavioural Sciences, Studies in Education*, 2016.

22 Magnus Nilsson. Swedish "immigrant literature" and the construction of ethnicity. *Tijdschrift voor skandinavistiek*, 31(1), 2010.

23 Katsuo A Nishikawa, Terri L Towner, Rosalee A Clawson, and Eric N Waltenburg. Interviewing the interviewers: Journalistic norms and racial diversity in the newsroom. *The Howard Journal of Communications*, 20(3):242–259, 2009.

24 Ansgar Nünning. Narratology or narratologies? taking stock of recent developments, critique and modest proposals for future usages of the term. *What Is Narratology? Questions and Answers Regarding the Status of a Theory*, pages 239–75, 2003.

25 Tim O'Gorman, Kristin Wright-Bettner, and Martha Palmer. Richer event description: Integrating event coreference with temporal, causal and bridging annotation. In *Proceedings of the 2nd Workshop on Computing News Storylines (CNS 2016)*, pages 47–56, 2016.

26 Michele Pasin and John Bradley. Factoid-based prosopography and computer ontologies: towards an integrated approach. *Digital Scholarship in the Humanities*, 30(1):86–97, 2015.

27 Valentina Presutti and Aldo Gangemi. Content ontology design patterns as practical building blocks for web ontologies. In *International Conference on Conceptual Modeling*, pages 128–141. Springer, 2008.

28 James Pustejovsky, José M Castano, Robert Ingria, Roser Sauri, Robert J Gaizauskas, Andrea Setzer, Graham Katz, and Dragomir R Radev. Timeml: Robust specification of event and temporal expressions in text. *New directions in question answering*, 3:28–34, 2003.

29 Lama Saeeda, Michal Med, Martin Ledvinka, Miroslav Blaško, and Petr Křemen. Entity linking and lexico-semantic patterns for ontology learning. In *European Semantic Web Conference*, pages 138–153. Springer, 2020.

30 Cogan Shimizu, Pascal Hitzler, Quinn Hirt, Dean Rehberger, Seila Gonzalez Estrecha, Catherine Foley, Alicia M Sheill, Walter Hawthorne, Jeff Mixter, Ethan Watrall, et al. The enslaved ontology: Peoples of the historic slave trade. *Journal of Web Semantics*, 63:100567, 2020.

31 John Simpson and Susan Brown. From xml to rdf in the orlando project. In *2013 International Conference on Culture and Computing*, pages 194–195. IEEE, 2013.

32 Zhiyi Song, Ann Bies, Stephanie Strassel, Tom Riese, Justin Mott, Joe Ellis, Jonathan Wright, Seth Kulick, Neville Ryant, and Xiaoyi Ma. From light to rich ere: annotation of entities, relations, and events. In *Proceedings of the the 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 89–98, 2015.

33 Gayatri Chakravorty Spivak. Can the subaltern speak? *Die Philosophin*, 14(27):42–58, 2003.

34 Cui Tao, Harold R Solbrig, and Christopher G Chute. Cntro 2.0: a harmonized semantic web ontology for temporal relation inferencing in clinical narratives. *AMIA summits on translational science proceedings*, 2011:64, 2011.

35 Cui Tao, Wei-Qi Wei, Harold R Solbrig, Guergana Savova, and Christopher G Chute. Cntro: a semantic web ontology for temporal relation inferencing in clinical narratives. In *AMIA annual symposium proceedings*, volume 2010, page 787. American Medical Informatics Association, 2010.

36 Jian-hua Yeh. Towards a biographic knowledge-based story ontology system. In *Proceedings of the 2018 International Conference on Intelligent Information Technology*, pages 33–38, 2018.

37 Amy Zhao Yu, Shahar Ronen, Kevin Hu, Tiffany Lu, and César A Hidalgo. Pantheon 1.0, a manually verified dataset of globally famous biographies. *Scientific data*, 3(1):1–16, 2016.

38 Lu Zhou, Cogan Shimizu, Pascal Hitzler, Alicia M Sheill, Seila Gonzalez Estrecha, Catherine Foley, Duncan Tarr, and Dean Rehberger. The enslaved dataset: A real-world complex ontology alignment benchmark using wikibase. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 3197–3204, 2020.

Plenary Debates of the Parliament of Finland as Linked Open Data and in Parla-CLARIN Markup

Laura Sinikallio
HELDIG Centre for Digital Humanities,
SeCo Research Group,
University of Helsinki, Finland

Senka Drobac
Department of Computer Science,
SeCo Research Group,
Aalto University, Finland

Minna Tamper
Department of Computer Science,
SeCo Research Group,
Aalto University, Finland

Rafael Leal
HELDIG Centre for Digital Humanities,
SeCo Research Group,
University of Helsinki, Finland

Mikko Koho
HELDIG Centre for Digital Humanities,
SeCo Research Group,
University of Helsinki, Finland

Jouni Tuominen
Aalto University, Finland
HELDIG Centre for Digital Humanities,
SeCo Research Group,
University of Helsinki, Finland

Matti La Mela
HELDIG Centre for Digital Humanities,
SeCo Research Group,
University of Helsinki, Finland

Eero Hyvönen
Aalto University, Finland
HELDIG Centre for Digital Humanities,
SeCo Research Group,
University of Helsinki, Finland

Abstract

This paper presents a knowledge graph created by transforming the plenary debates of the Parliament of Finland (1907–) into Linked Open Data (LOD). The data, totaling over 900 000 speeches, with automatically created semantic annotations and rich ontology-based metadata, are published in a Linked Open Data Service and are used via a SPARQL API and as data dumps. The speech data is part of larger LOD publication *FinnParla* that also includes prosopographical data about the politicians. The data is being used for studying parliamentary language and culture in Digital Humanities in several universities. To serve a wider variety of users, the entirety of this data was also produced using Parla-CLARIN markup. We present the first publication of all Finnish parliamentary debates as data. Technical novelties in our approach include the use of both Parla-CLARIN and an RDF schema developed for representing the speeches, integration of the data to a new Parliament of Finland Ontology for deeper data analyses, and enriching the data with a variety of external national and international data sources.

2012 ACM Subject Classification Information systems → Ontologies; Information systems → Resource Description Framework (RDF); Computing methodologies → Information extraction

Keywords and phrases Plenary debates, parliamentary data, Parla-CLARIN, Linked Open Data, Digital Humanities

Digital Object Identifier 10.4230/OASIcs.LDK.2021.8

Acknowledgements Thanks to Ari Apilo, Sari Wilenius, and Päivikki Karhula of PoF for providing material for the project. Our work was funded by the Academy of Finland as part of the Semantic Parliament project, the EU project InTaVia: In/Tangible European Heritage[1], and is related to the COST action NexusLinguarum[2] on linguistic data science. CSC – IT Center for Science, Finland, provided computational resources for the work.

[1] `https://intavia.eu`
[2] `https://nexuslinguarum.eu`

1 Introduction

Semantic Parliament (SEMPARL)[3] is a consortium research project, which produces a linked open data and research infrastructure on Finnish parliamentary data, and develops novel semantic computing technologies to study parliamentary politics and political culture. SEMPARL brings together researchers at the University of Helsinki, University of Turku, and Aalto University, with complementary, multi-disciplinary expertise in language technology, political and media research, and semantic computing and web technologies, respectively.

The project makes three major contributions. First, it responds to the demand for an easy to use and "intelligent" access to the newly digitized Finnish parliamentary data by providing the data as a national Linked Open Data (LOD) infrastructure and service for researchers, citizens, the government, and the media, and application developers. Second, the project studies long-term changes in the Finnish parliamentary and political culture and language. These use cases in political and language research are pioneering studies using the Finnish digital parliamentary data. Third, the new data service semantically enriches content in other related Finnish LOD services, such as LawSampo for Finnish legislation and case law [7] and BiographySampo for prosopographical data [6].

From a Linked Data production point of, two interlinked knowledge graphs (KG) are produced in SEMPARL: 1) A KG of all over 900 000 parliamentary debate speeches of the Parliament of Finland (PoF) (1907–present) to be called S-KG. 2) A prosopographical knowledge (P-KG) graph of the over 2600 Members of Parliament (MP), other people, and organizations related to the parliamentary speeches during the same period of time [16]. These KGs constitute together a larger data publication of PoF data called FinnParla. This paper presents the first graph S-KG and addresses the following more general research question: *How to represent and publish parliamentary speeches so that the data can be used easily for Digital Humanities research?*

In the following, we first present the problem of representing publishing, and using plenary debates as data for Digital Humanities research, and discuss related works and projects. After this, our original debate data, target data model, and the transformation process are described. The produced linked data has been published as a data service using the 7-star model of the Linked Data Finland platform [8]. As a demonstration of using the data service in Digital Humanities research, exemplary data-analyses are presented using YASGUI and Google Colab on top of the underlying SPARQL endpoint. In conclusion, contributions of the work are summarized, related works are discussed, and further research are outlined.

2 Related Work: Publishing Plenary Debates as Data

The Unicameral Parliament of Finland convened for the first time in 1907. The parliament has 200 members (MP), who are elected for four years. Since the first parliament of 1907, the elections are based on universal suffrage and both male and female MPs have been elected to all parliaments. In the Finnish parliament, the debates take place in the public plenary sessions. Since 1907, the Parliament has transcribed the speeches and published the printed plenary session minutes, which is a practice established already in the nineteenth-century Diet of Estates [20]. The minutes contain the matters considered, the decisions made, and every speech heard during the sessions. The wordings of the speeches are revised and improved for readability [29, 20].

[3] `https://seco.cs.aalto.fi/projects/semparl/en/`

In the 1990s, the Parliament of Finland started to gradually publish parliamentary documents in digital form. It was only in 2018 that the Parliament completed the digitisation of the historical parliamentary documents of 1907–1999 and opened a new version of their data service [10]. This open data and the data service of the Parliament, however, has weaknesses concerning the data and its usability due to the heterogeneous data formats and different ways of access. For example, the historical minutes contain only the text recognised from the image files, and have no metadata concerning the structure of the minutes or their content, which limits the research to bag-of-words approaches [14].

There are also annotated corpora produced of the Finnish Parliamentary debates, which cover the recent decades. FIN-CLARIN has a curated corpus of the debates in 2008–2016 [3]. These include linguistic annotation, metadata about the speakers and the speeches are linked to the actual video recordings of the plenary sessions. Moreover, there is the multilingual Parlspeech parliamentary corpus [21], which includes also the plenary debates of the Finnish parliament in 1991–2015. This data, however, has quality problems. It has been created from the PDF files of the Parliament website of the time, but not all the speeches can be found in the data when we compare it with the complete minutes.

Several projects have transformed parliamentary debates into structured data or produced annotated parliamentary debate corpora. Regarding the former, the projects have foremost concerned the digitisation of the parliamentary debates and their enrichment with political or biographical metadata. These data have been transformed both to XML and RDF format[4]. In the Lipad project, the Canadian Hansard from 1901 to present was transformed into linked XML structured data [1]. As in our case, the process included both the OCR and the parsing of the historical documents and more straight-forward conversion of the recent SQL parliamentary data. The major example of parliamentary data in RDF is the Linked EP project, where the data of the European parliament 1999–2017 was transformed into RDF format and enriched with biographical information [28]. The RDF standard has been used also in the Latvian LinkedSAEIMA project [2], in the Italian Parliament[5] and in the PoliMedia project, where RDF parliamentary data was linked with media sources [11].

There are several parliamentary corpora. The best known is perhaps the EuroParl corpus, which includes the plenary session debates of the European Parliament and has been used to study machine translation [12]. A comprehensive list of the national parliamentary corpora is presented on the CLARIN webpage[6]. The Talk of Norway (1998–2016) is an example of a national parliament corpus with linguistic annotation published in CSV and TSV formats [15]. Different guidelines have been followed for annotating and encoding the Parliamentary debates. The TEI-based Parla-CLARIN schema, which we also use in our transformation, is an attempt to define a common annotation model.[7] For example, the Slovene parliamentary corpus siParl (1990–2018) has been encoded with the Parla-CLARIN schema [19]. Currently, the Parla-CLARIN schema is implemented in the Clarin ParlaMint project[8], which establishes a comparable and interoperable corpus of almost twenty national parliamentary corpora for comparative research.

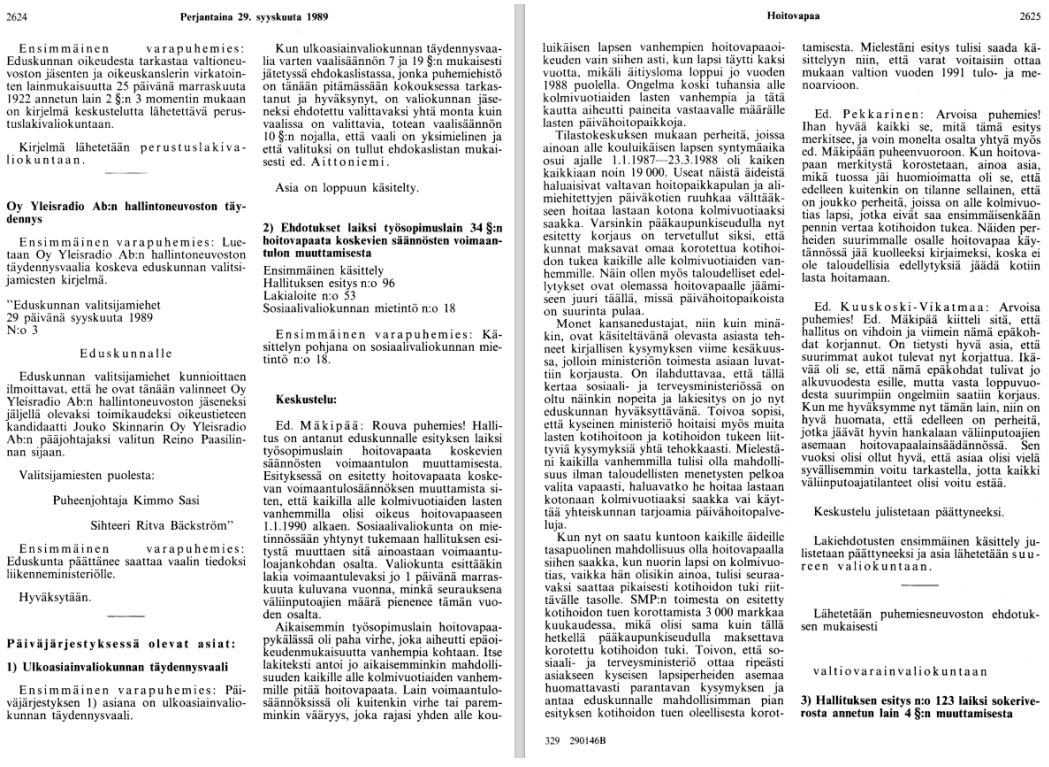
A novelty in the transformation done in our SEMPARL project is to combine RDF standard with Parla-CLARIN schema. Moreover, most of the annotated parliamentary corpora cover mainly the recent years while in our case the complete work of the PoF from 1907 is covered – and for the first time.

[4] `https://www.w3.org/RDF/`
[5] `http://data.camera.it/data/en/datasets/`
[6] `https://www.clarin.eu/resource-families/parliamentary-corpora`
[7] See: `https://www.clarin.eu/blog/clarin-parlaformat-workshop`
[8] `https://github.com/clarin-eric/ParlaMint`

3 Original Data

The original data, minutes of Finnish plenary sessions, was gathered from several sources and in three different formats depending on the availability: 1) From 1907 to 1999[9] the plenary session minutes are available only in PDF format[10]. One parliamentary session is split into 1–8 separate PDF files, each containing the minutes for several plenary sessions. 2) From halfway parliamentary session 1999 to the end of session 2014, the data is available also in HTML format at PoF's web pages[11]. 3) From session 2015 onward the plenary sessions are available as XML from the *Avoin eduskunta* API[12].

Figure 1 shows an example of original PDF-format minutes for plenary session 87/1989[13]. Later minutes available in HTML and XML also mostly follow shown layout and logic; In general, the minutes consist of items (or topics), marked here in bold (except the row *Keskustelu:*). The item header is followed by: a possible list of related documents, chairman's opening comments, a possible debate section marked by *Keskustelu:* (*debate/conversation*) and finally a decision and a closing statement.



2624 Perjantaina 29. syyskuuta 1989

Ensimmäinen varapuhemies: Eduskunnan oikeudesta tarkastaa valtioneuvoston jäsenten ja oikeuskanslerin virkatointen lainmukaisuutta 25 päivänä marraskuuta 1922 annetun lain 2 §:n 3 momentin mukaan on kirjelmä keskustelutta lähetettävä perustuslakivaliokuntaan.

Kirjelmä lähetetään perustuslakivaliokuntaan.

Oy Yleisradio Ab:n hallintoneuvoston täydennys

Ensimmäinen varapuhemies: Luetaan Oy Yleisradio Ab:n hallintoneuvoston täydennysvaalia koskeva eduskunnan valitsijamiesten kirjelmä.

"Eduskunnan valitsijamiehet
29 päivänä syyskuuta 1989
N:o 3

Eduskunnalle

Eduskunnan valitsijamiehet kunnioittaen ilmoittavat, että he ovat tänään valinneet Oy Yleisradio Ab:n hallintoneuvoston jäseneksi jäljellä olevaksi toimikaudeksi oikeustieteen kandidaatti Jouko Skinnarin Oy Yleisradio Ab:n pääjohtajaksi valitun Reino Paasilinnan sijaan.

Valitsijamiesten puolesta:

Puheenjohtaja Kimmo Sasi

Sihteeri Ritva Bäckström"

Ensimmäinen varapuhemies: Eduskunta päättänee saattaa vaalin tiedoksi liikenneministeriölle.

Hyväksytään.

Päiväjärjestyksessä olevat asiat:

1) Ulkoasiainvaliokunnan täydennysvaali

Ensimmäinen varapuhemies: Päiväjärjestyksen 1) asiana on ulkoasiainvaliokunnan täydennysvaali.

Kun ulkoasiainvaliokunnan täydennysvaalia varten vaalisäännön 7 ja 19 §:n mukaisesti jätetyssä ehdokaslistassa, jonka puhemiehistö on tänään pitämässään kokouksessa tarkastanut ja hyväksynyt, on valiokunnan jäseneksi ehdotettu valittavaksi yhtä monta kuin vaalissa on valittavia, totean vaalisäännön 10 §:n nojalla, että vaali on yksimielinen ja että valituksi on tullut ehdokaslistan mukaisesti ed. Aittoniemi.

Asia on loppuun käsitelty.

2) Ehdotukset laiksi työsopimuslain 34 §:n hoitovapaata koskevien säännösten voimaantulon muuttamisesta

Ensimmäinen käsittely
Hallituksen esitys n:o 96
Lakialoite n:o 53
Sosiaalivaliokunnan mietintö n:o 18

Ensimmäinen varapuhemies: Käsittelyn pohjana on sosiaalivaliokunnan mietintö n:o 18.

Keskustelu:

Ed. Mäkipää: Rouva puhemies! Hallitus on antanut eduskunnalle esityksen laiksi työsopimuslain hoitovapaata koskevien säännösten voimaantulon muuttamisesta. Esityksessä on esitetty hoitovapaata koskevan voimaantulosäännöksen muuttamista siten, että kaikilla alle kolmivuotiaiden lasten vanhemmilla olisi oikeus hoitovapaaseen 1.1.1990 alkaen. Sosiaalivaliokunta on mietinnössään yhtynyt tukemaan hallituksen esitystä muuttaen sitä ainoastaan voimaantuloajankohdan osalta. Valiokunta esittääkin lakia voimaantulevaksi jo 1 päivänä marraskuuta kuluvana vuonna, minkä seurauksena väliinputoajien määrä pienenee tämän vuoden osalta.

Aikaisemmin työsopimuslain hoitovapaapykälässä oli paha virhe, joka aiheutti epäoikeudenmukaisuutta vanhempia kohtaan. Itse lakiteksti antoi jo aikaisemminkin mahdollisuuden kaikille alle kolmivuotiaiden vanhemmille pitää hoitovapaata. Lain voimaantulosäännöksissä oli kuitenkin virhe tai paremminkin vääryys, joka rajasi yhden alle kouluikäisen lapsen vanhempien hoitovapaaoikeuden vain siihen asti, kun lapsi täytti kaksi vuotta, mikäli äitiysloma loppui jo vuoden 1988 puolella. Ongelma koski tuhansia alle kolmivuotiaiden lasten vanhempia ja tätä kautta aiheutti paineita vastaavalle määrälle lasten päivähoitopaikkoja.

Tilastokeskuksen mukaan perheitä, joissa ainoan alle kouluikäisen lapsen syntymäaika osui ajalle 1.1.1987—23.3.1988 oli kaiken kaikkiaan noin 19 000. Useat näistä äideistä haluaisivat valtavan hoitopaikkapulan ja alimiehitettyjen päiväkotien ruuhkaa välttääkseen hoitaa lastaan kotona kolmivuotiaaksi saakka. Varsinkin pääkaupunkiseudulla nyt esitetty korjaus on tervetullut siksi, että kunnat maksavat omaa korotettua kotihoidon tukea kaikille alle kolmivuotiaiden vanhemmille. Näin ollen myös taloudelliset edellytykset ovat olemassa hoitovapaalle jäämiseen juuri täällä, missä päivähoitopaikoista on suurinta pulaa.

Monet kansanedustajat, niin kuin minäkin, ovat käsiteltävänä olevasta asiasta tehneet kirjallisen kysymyksen viime kesäkuussa, jolloin ministeriön toimesta asiaan luvattiin korjausta. On ilahduttavaa, että tällä kertaa sosiaali- ja terveysministeriössä on oltu näinkin nopeita ja lakiesitys on jo nyt eduskunnan hyväksyttävänä. Toivoa sopisi, että kyseinen ministeriö hoitaisi myös muita lasten kotihoitoon ja kotihoidon tukeen liittyviä kysymyksiä yhtä tehokkaasti. Mielestäni kaikilla vanhemmilla tulisi olla mahdollisuus ilman taloudellisten menetysten pelkoa valita vapaasti, haluavatko he hoitaa lastaan kotonaan kolmivuotiaaksi saakka vai käyttää yhteiskunnan tarjoamia päivähoitopalveluja.

Kun nyt on saatu kuntoon kaikille äideille tasapuolinen mahdollisuus olla hoitovapaalla siihen saakka, kun nuorin lapsi on kolmivuotias, vaikka hän olisikin ainoa, tulisi seuraavaksi saattaa pikaisesti kotihoidon tuki riittävälle tasolle. SMP:n toimesta on esitetty kotihoidon tuen korottamista 3 000 markkaa kuukaudessa, mikä olisi sama kuin tällä hetkellä pääkaupunkiseudulla maksettava korotettu kotihoidon tuki. Toivon, että sosiaali- ja terveysministeriö ottaa ripeästi asiakseen kyseisen lapsiperheiden asemaa huomattavasti parantavan kysymyksen ja antaa eduskunnalle mahdollisimman pian esityksen kotihoidon tuen oleellisesta korotamisesta. Mielestäni esitys tulisi saada käsittelyyn niin, että varat voitaisiin ottaa mukaan valtion vuoden 1991 tulo- ja menoarvioon.

Ed. Pekkarinen: Arvoisa puhemies! Ihan hyvää kaikki se, mitä tämä esitys merkitsee, ja voin monelta osalta yhtyä myös ed. Mäkipään puheenvuoroon. Kun hoitovapaan merkitystä korostetaan, ainoa asia, mikä tuossa jäi huomioimatta oli se, että edelleen kuitenkin on tilanne sellainen, että on joukko perheitä, joissa on alle kolmivuotias lapsi, jotka eivät saa ensimmäisenkään pennin vertaa kotihoidon tukea. Näiden perheiden suurimmalle osalle hoitovapaa käytännössä jää kuolleeksi kirjaimeksi, koska ei ole taloudellisia edellytyksiä jäädä kotiin lasta hoitamaan.

Ed. Kuuskoski-Vikatmaa: Arvoisa puhemies! Ed. Mäkipää kiitteli sitä, että hallitus on vihdoin ja viimein nämä epäkohdat korjannut. On tietysti hyvä asia, että suurimmat aukot tulevat nyt korjattua. Ikävää oli se, että nämä epäkohdat tulivat jo alkuvuodesta esille, mutta vasta loppuvuodesta suurimpiin ongelmiin saatiin korjaus. Kun me hyväksymme nyt tämän lain, niin on hyvä huomata, että edelleen on perheitä, jotka jäävät hyvin hankalaan väliinputoajien asemaan hoitovapaalainsäädännössä. Sen vuoksi olisi ollut hyvä, että asiaa olisi vielä syvällisemmin voitu tarkastella, jotta kaikki väliinputoajatilanteet olisi voitu estää.

Keskustelu julistetaan päättyneeksi.

Lakiehdotusten ensimmäinen käsittely julistetaan päättyneeksi ja asia lähetetään suureen valiokuntaan.

Lähetetään puhemiesneuvoston ehdotuksen mukaisesti

valtiovarainvaliokuntaan

3) Hallituksen esitys n:o 123 laiksi sokeriverosta annetun lain 4 §:n muuttamisesta

Figure 1 Example of a plenary session transcript. Available by the CC BY 4.0 licence.

[9] There is no data for 1915 and 1916 as due to war the Parliament did not convene.
[10] https://avoindata.eduskunta.fi/#/fi/digitoidut/download
[11] https://www.eduskunta.fi/FI/taysistunto/Sivut/Taysistuntojen-poytakirjat.aspx
[12] https://avoindata.eduskunta.fi/#/fi/home
[13] https://s3-eu-west-1.amazonaws.com/eduskunta-asiakirja-original-documents-prod/suomi/1989/PTK_1989_3.pdf

Each source format differs in the metadata included. All formats contained the essential data, such as plenary session id, date, debate topic, speaker's last name, and role. The newer machine readable formats have been enriched with additional data, such as URLs to documents related to the debate topics or even individual starting and ending times for a speech. Table 1 illustrates the metadata present in each format and distribution of used source formats.

Table 1 Distribution of used source data format and variant metadata present in it. Row *Ubiquitous metadata* lists metadata that was available in all formats. * HTML became available after plenary session 85/1999.

	Parliamentary session	Speaker first name	Speaker party	Item transcript URL	Related document URL	Session transcript status	Session transcript version	Speech transcript status	Speech transcript version	Speech start time	Speech end time
PDF	1907-1999*	-	-	-	-	-	-	-	-	-	-
HTML	1999*-2014	X	X	X	X	-	-	-	-	-	-
XML	2015-2020	X	X	-	-	X	X	X	X	X	X
Ubiquitous metadata		session date, session ending and starting times, session id, speaker last name, speaker title, speech type, related documents, debate topic									

4 Target Data Model

The goal of the whole data transformation process was to make all data available in a coherent, unified format. In this project we did this twice-fold in Parla-CLARIN XML and RDF. The central unit of the data is a speech; any comment, statement or vocal contribution made during a plenary session[14]. The goal of the transformation process was to find all such speeches and all available metadata related to them. Generally we refer to all before-mentioned instances as speeches. For full coverage we have also gathered all speeches made by the chairmen. These are mostly about guiding the progression of a session.

The Parla-CLARIN XML format[15] for representing speech texts is an easily readable chronological presentation of the debate data for both machines and humans. We produced one file per parliamentary session. Listing 1 gives an example of a section from the final data in Parla-CLARIN XML. The excerpt covers the start of the debate on a topic during the plenary session 37/2005.

By transforming all data to RDF as well, we aimed to create the knowledge graph (S-KG) of all parliamentary debate speeches. For this purpose a customised RDF-based metadata schema was created. The schema contains six different, interlinked classes: Speech, Interruption, Item, Session, Document, and Transcript. Speeches were represented as

[14] These do not include interjections, other vocal interruptions or chairman comments made during a speech. In original data these have been embedded into the actual speeches. These were handled in the transformation process as *interruptions*.

[15] `https://clarin-eric.github.io/parla-clarin/`

instances of the class Speech with 24 properties (metadata elements) as described in Table 2. Here the default namespace is our own (*semparls*); *bioc* refers to the BioCRM schema for representing biographical data [27]; *rdfs* refers to the RDFS Schema and *xsd* to the XML Schema of W3C. The column C tells the cardinality of the property, Range the range, and last column the meaning of the property. Table 3 describes in the same way the remaining five classes and additionally a seventh class, NamedEntity, that was created by post-transformation language analysis.

Listing 1 An abridged excerpt from the Parla-CLARIN data.

```
<TEI xml:id="ptk_37_2005">
  [...]
   <div>
   <head>
      Eduskunnan pankkivaltuuston kertomus 2014
      <listBibl>
        <head>Related documents:</head>
        <bibl>Kertomus K 14/2015 vp</bibl>
      </listBibl>
   </head>
   <div>
      <note link=[...] speechType="" type="speaker" xml:id="2015.24.102"/>
      <u ana="#secondViceChair" who="#Paula_Risikko" xml:id="2015.24.102">
      Lähetekeskustelua varten esitellään päiväjärjestyksen 4. asia.
      Puhemiesneuvosto ehdottaa, että asia lähetetään talousvaliokuntaan
      Meille asian esittelee edustaja Zyskowicz, olkaa hyvä.</u>
   </div>
   <div>
      <note end="2015-06-24T17:54:02" link=[...] speechType="Esittelypuheenvuoro"
      start="2015-06-24T17:45:01" type="speaker" xml:id="2015.24.103"/>
      <u who="#Ben_Zyskowicz" xml:id="2015.24.103">Arvoisa rouva puhemies!
      Arvoisat kansanedustajat! Käsittelyssä on nyt Pankkivaltuuston kertomus
      vuodelta 2014. Kuten viime vuonnakin, [...] Loppuosa eli noin 137,5
      iljoonaa euroa siirrettiin valtion loputtomiin tarpeisiin.</u>
   </div>
   <div>
      <note end="2015-06-24T18:02:27" link=[...] speechType=""
      start="2015-06-24T17:54:07" type="speaker" xml:id="2015.24.104"/>
      <u next="2015.24.104.2" who="#Olavi_Ala-Nissilä" xml:id="2015.24.104.1">
      Arvoisa rouva puhemies! Tässä entinen Pankkivaltuuston puheenjohtaja,
      nykyinen jäsen, edustaja Zyskowicz käytti hyvän puheenvuoron. [...]
      Muistan, kun silloin valtiovarainministeri ja ministeri Wideroos </u>
      <vocal who="Eero_Heinäluoma">
          <desc>Eero Heinäluoma: Toinen valtiovarainministeri!</desc>
      </vocal>
      <u prev="2015.24.104.1" who="#Olavi_Ala-Nissilä" xml:id="2015.24.104.2">
      – toinen valtiovarainministeri Wideroos – ja hallituskin ajoivat sitä,
      että Suomen Pankin pääomia [...] ja Kreikan on omankin taloutensa
      kannalta välttämättä saatava julkinen hallintonsa paremmin toimimaan.</u>
      <vocal>
          <desc>Eduskunnasta: Hyvä puheenvuoro!</desc>
      </vocal>
   </div>
[...]
```

The data model presented for representing debates is part of a larger Ontology of Parliament of Finland under development in the SEMPARL project. This ontology is based on the CIDOC CRM[16]-based Bio CRM model [27], where parliamentary events are represented in time and place with actors (people, groups, such as parties, and organizations) participating in different roles. The ontology is populated with data extracted from the speech data and databases of PoF [16]. For example, the *:speaker* and *:party* property values in Table 2 are filled with resources taken from the actor graph in the PoF ontology that contains over 2600 MPs, ministers, presidents of Finland, and other prominent people related to the speeches as speakers or mentioned in the texts. In this way, prosopographical data and the speeches can be integrated seamlessly and be used together with the Digital Humanities

[16] `http://cidoc-crm.org`

Table 2 Semparls RDF schema for Speech. [a]From some source data the chairmen names were not always reliably recognizable. In this case chairman speeches lack this value.

Speech			
Element URI	**C**	**Range**	**Meaning of the value**
:skos:prefLabel	1	rdf:langString	String label for speech
:speaker	0..1[a]	bioc:Person	Person speaking URI
:party	0..1	:Party	Party of the speaker URI
:partyInSource	0..1	rdfs:Literal	Party as written in the source if available
:role	1	:Role	Speaker's role
:speakerInSource	1	rdfs:Literal	Speaker's name as in source
:speechOrder	1	xsd:integer	Ordinal of the speech in a session
:content	1	rdfs:Literal	Speech as text (incl. interruptions)
dct:language	0..*	rdfs:Resource	Recognized languages of the speech
:speechType	0..1	:SpeechType	Type of the speech
:isInterruptedBy	0..*	:Interruption	Interruptions during the speech
dct:date	1	xsd:date	Date of the session
:startTime	0..1	xsd:time	Start time of the speech
:endDate	0..1	xsd:date	Session end date if not same as date
:endTime	0..1	xsd:time	End time of the speech
:item	0..1	:Item	Item in agenda/topic of the speech
:session	1	:Session	Session where the speech was made
:diary	1	rdfs:Resource	URL of session transcript
:page	0..1	xsd:integer	Page number for PDF-based data
:status	0..1	:Status	Status of the speech transcription
:version	0..1	xsd:decimal	Version of the speech transcription
:namedEntity	0..*	:NamedEntity	Referenced named entities
dct:subject	0..*	skos:Concept	Subject matter keywords

analyses of the parliamentary data. For example, by using biographical information about the speaker it is possible to investigate how much (s)he has spoken about matters related to his/her own electoral district.

5 Transformation Process

Semantic Parliament aggregates data from several disparate source databases into a unified knowledge graph. An overall plan of the data transformation processes of source datasets and the linking of entities between different parts are shown in Figure 2. The source datasets are shown as rectangles on the left side of the transformation pipeline and the RDF-format parts are shown as yellow cylinders. The solid arrows depict data transformation and dotted arrows correspond to entity linking either inside the Semantic Parliament data or to external ontologies and datasets (shown on the top).

The external ontologies and data shown in Figure 2 are the AMMO ontology of Finnish historical occupations, which is linked to social statuses through the international HISCO standard [13], Wikidata, related Finnish Sampo data services and portals[17], such as LawSampo [7] and BiographySampo [6], places, Finto[18] ontologies, EKS subject headings[19] used in the library of PoF, Semantic Finlex [18] data service of Finnish legislation and case law [18], and

[17] https://seco.cs.aalto.fi/events/2020/2020-10-29-sampo-portals/
[18] https://finto.fi/en/
[19] https://www.eduskunta.fi/kirjasto/EKS/index.html?kieli=en

Table 3 Semparls RDF schema for the classes Interruption, Item, Document, Session, Transcript, and ReferencedNamedEntity. Each class also contains the predicate *skos:prefLabel* that has been omitted from the table for redundancy.

Element URI	C	Range	Meaning of the value
Interruption			
:content	1	rdfs:Literal	Content of the interruption
:interrupter	0..1	rdfs:Literal	Source of the interruption
:speaker	0..1	bioc:Person	Interrupter URI, if interrupter was mentioned
Item			
:session	1	:Session	Session where item on agenda
dct:title	1	rdf:langString	Title as written in source
:relatedDocument	0..*	:Document	Document related to item
:diary	1	rdfs:Resource	URL to online transcript
Document			
dct:title	1	xsd:string	Name of the document
:id	0..1	xsd:string	Official Parliament id
:url	0..1	rdfs:Resource	URL to online transcript
Session			
:id	1	rdfs:Literal	Session id/ session number
dct:date	1	xsd:date	Date of the session
:startTime	0..1	xsd:time	Start time of the session
:endDate	0..1	xsd:date	Session end date if not same as session date
:endTime	0..1	xsd:time	End time of the session
:transcript	1	:Transcript	Transcript of the session
Transcript			
:status	0..1	:Status	Status of the transcript
:version	0..1	xsd:decimal	Version of the transcript
:url	1	rdfs:Resource	URL to online transcript
NamedEntity			
:surfaceForm	1	xsd:string	original surface forms in text
:count	1	xsd:integer	how many times entity is mentioned in a speech
:category	1	xsd:string	type of the named entity
:surfaceForm	1	xsd:string	named entity in surface form
skos:relatedMatch	0..*	rdfs:Resource	links to ontologies for named entities

the Lakitutka[20] service publishing data related to government proposals discussed in the speeches and other documents. These will enrich the content and enhance the usefulness of the speech data for parliamentary research and applications.

The step 1 of transforming MP data is discussed in [16]. The step 2 concerning government proposals remains a future work. This paper focuses on the 3. step of the transformation of the plenary session documents and the full-text contents of the speeches given in the sessions. The entity linking from the plenary sessions to entities of the MP data is already implemented, as well as linking to places, Finto ontologies and Semantic Finlex, while linking to government proposal documents, EKS, and Lakitutka will be implemented in the future.

OCR Process In the 3. step, the data from 1907 until 1999 was available only as scanned images combined into PDF files, which needed to be first processed into machine-readable text. The quality of the scanned documents is generally good, with older documents having partially smudged parts of the text and some pages slightly skewed. The text in the documents

[20] `https://lakitutka.fi`

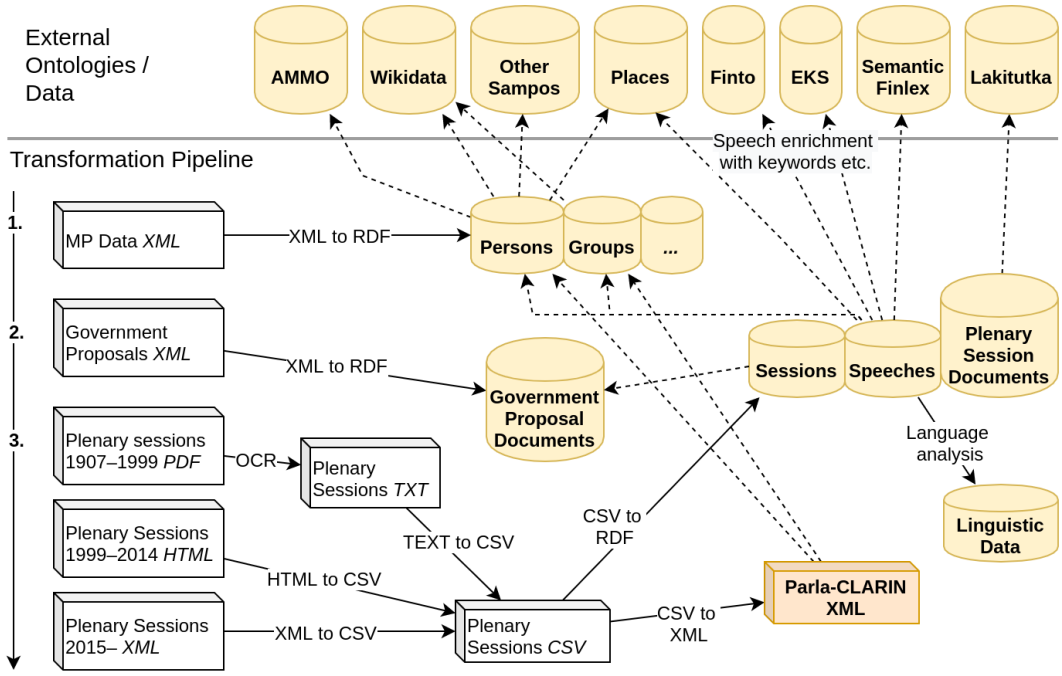


Figure 2 Transformation process and source datasets of Semantic Parliament.

is formatted into two columns, with older issues separated with a black line. There is a difference in the fonts used in different years. However, both early and later years are printed with modern fonts that are easy to recognize. Most of the text is written in Finnish, however, there are some parts written in Swedish (another official language of Finland), so we needed to use a multilingual OCR model for recognition.

For the OCR, we used Tesseract 5[21], with the default Finnish and Swedish models together for recognition `fin+swe`. The initial experiments showed that Tesseract's pre-trained models worked well with our data so we didn't need to create any training data and train new models, which simplified the whole process. Also, Tesseract's possibility to use multi-model recognition was very convenient for our dataset. As the output from the OCR process, we opted for the plain text as it seemed to be more convenient for further processing.

Since the scanned images are available in PDF files, to OCR them we needed to first transform them to PNG format. We performed the transformation with `pdftopng` program with 350 dpi resolution. In the initial experiments, we tried the OCR process with different resolutions, but the 350 dpi seemed to give the best results with pre-trained OCR models.

The quality of the OCR seems to be generally good enough for our purpose. We have noticed that there are lots of mistakes in tables and lists due to Tesseract's segmentation problems. But, since we are focusing only on extracting parliamentary discussions, which are contained in the running text, we are satisfied with the OCR quality. However, during the processing of the data, we did perform some post-correction, like removing extra characters and end-of-line hyphenation, and correction of speaker names and headers.

Gathering and editing the data. For the OCR-based data we decided to add one manual step to the process. Every plenary session's original minutes start with a clearly structured header row containing central information about the session (i.e. session number and date).

[21] `https://github.com/tesseract-ocr`, version: 5.0.0-alpha-648-gcdebe

Where the rest of the document was in most cases laid out into two columns, this header spanned both columns and was hence occasionally split or otherwise corrupted in the OCR process. To considerably improve the reliability of this central metadata, we chose to go through the files with the help of a printer script to spot these mangled headers and manually fix them. After that all relevant data was gathered with the use of regular expressions.

For the HTML-based data (step 3 in Fig. 2), we needed two steps to gather all the data. The HTML-based minutes were separated into a) a main page, listing the agenda, and links to possible debate pages and related documents, and b) possible debate pages that contained the actual debate related to an item on the agenda. Gathering the data required first scraping the main pages and then, based on the discussion page links found, the discussion data. Finally data from these sources needed to be reordered and combined into an integrated whole.

The XML-based data (2015–) was gathered with requests to *Avoin eduskunta* API that returned the minutes as JSON-wrapped XML data. The HTML- and XML-based data consisted of pre-processed elements and was mostly quite ready to use as it is. For HTML some elements did require a few string operations to split information for separate values. Regardless of the original format, all data was first transformed into CSV format, one parliamentary session a file and one speech per row with columns representing the properties of the speeches. A unique ID was created for every speech in the process.

During the history of PoF there have been cases where two parliamentary sessions refer to the same calendar year. This is due to the government resigning in the middle of a parliamentary session and hence ending the session prematurely. For example, there was the first parliamentary session in 1975 and the second parliamentary session 1975 as well. Speech and plenary session IDs related to a second parliamentary session have a *_II* suffix attached. From the year 1917 we also transformed two unofficial but historically significant meetings that took place between parliamentary sessions. These speeches, sessions, and the files containing them are marked with a *_XX* suffix.

During editing and post-correction the speeches were cleaned of original end-of-line hyphenation and other unwanted characters but the original paragraph structure was kept. The clean-up results are not yet fully perfect but already usable. Some problems, like the occasional page header texts (that have carried over from the PDF based data) remain embedded in the speech content. Post-correction was also needed for two other notable issues that, however, only concerned the PDF-based data: 1) There are cases where the speeches had been wrongly split into two with the last section having incorrect metadata. 2) Speakers who had not been recognised in the data enrichment step (to be described below in more detail) are lacking in the metadata. This was either due to the speaker's name having been corrupted in some way during the process or (more rarely) due to that the person or certain form of their name is missing from the enrichment data source or original source deviating from typical transcript convention. The aim of post-correction was to automatically spot and fix such cases.

Data enrichment. During the transformations into CSV the data went through many post-corrections but also data enrichments. Most notably information about the speaker was expanded using the PoF Ontology. Where not already available in the original source, we fetched from the ontology the speaker's first name and party. If not already available in source material, we also automatically created URLs for relevant documents, such as original transcripts and related documents (bills, committee reports, etc.) if such existed. Language of each speech was checked with the LAS[22] tool.

[22] `http://demo.seco.tkk.fi/las/`

In order to analyze the speeches and to be able to study them in more detail, the named entities in the speeches were extracted and linked to the PoF Ontology (property :referencedNamedEntity in Table 2). In order to identify named entities from the speeches, the data had to be modeled to preserve structure and interjections within the texts. The speeches were transformed into RDF, using the NIF format[23] for interoperability, separating paragraphs and titles. The interjections were identified and marked as paragraphs, so that they could be extracted from the speeches themselves. After the separation process, the data can be used for morphological analysis on the speeches and interjections separately to enable text analysis. This, however, remains as a future work.

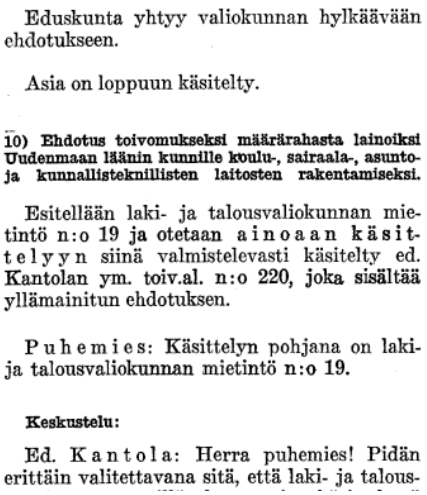
After the speeches were transformed into RDF to preserve their structure and to separate the speeches from interjections, the RDF was used to identify named entities from the texts. The named entity extraction was done using the upgraded Nelli tool [25] and linked separately to be able to take the context into account. The named entities (e.g., people, places, groups and organizations) were linked internally using the ARPA tool [17], in addition to resources in external knowledge bases, such as the Kanto[24] vocabulary for Finnish actors provided by the National Library for organizations and groups, the General Finnish Ontology (YSO) for places[25] [23], PNR[26] gazetteer data of Finnish place names by the National Survey, and the Semantic Finlex[27] [18] data of the Ministry of Justice to have broader coverage for linking places, actors, and legal documents.

The subject matter keywords for each speech were extracted using Annif [24], a subject indexing tool developed by the National Library of Finland (property *dct:subject* in Table 2). The Finto REST API[28] offers Annif models that are pre-trained on categorical metadata from Finnish libraries, museums, and archives available at the Finna service[29]. These projects provide subject keywords automatically linked to entities of the General Finnish Ontology YSO. The model used for subject indexing was `yso-fi`, which combines lexical and associative approaches, so that it is able to find terms directly present in the texts as well as indirect concepts based on statistical machine learning. A list of keywords for each speech was obtained using a limit of 100 keywords and a weight threshold of 0.01.

Parla-CLARIN Transformation. The transformation to Parla-CLARIN was a fairly straightforward process of creating an XML tree from the CSV data. Each file, containing one parliamentary session, forms its own entity, containing all session and speaker metadata with proper ID-linkage inside the document. We chose to separate all interruptions from the actual speech content by separating them to their own elements (as seen in Listing 1).

RDF Transformation. From the initial CSV, the debates were also transformed into RDF. For this we used the Terse RDF Triple Language (Turtle) syntax[30] and the schema presented in Section 4. The data for one parliamentary session was recreated as three different interlinked files, the first containing all the actual speeches made during that whole parliamentary session and all immediate metadata such as information about the speaker and the date. These link to a second file containing all the items discussed and related documents and their available

[23] `https://persistence.uni-leipzig.org/nlp2rdf/`
[24] `https://finto.fi/finaf/en/`
[25] `https://finto.fi/yso-paikat/en/?clang=en`
[26] `http://www.ldf.fi/dataset/pnr`
[27] `https://data.finlex.fi`
[28] `http://api.finto.fi/`
[29] `http://www.finna.fi`
[30] `https://www.w3.org/TR/turtle/`



Eduskunta yhtyy valiokunnan hylkäävään ehdotukseen.

Asia on loppuun käsitelty.

10) Ehdotus toivomukseksi määrärahasta lainoiksi Uudenmaan läänin kunnille koulu-, sairaala-, asunto- ja kunnallisteknillisten laitosten rakentamiseksi.

Esitellään laki- ja talousvaliokunnan mietintö n:o 19 ja otetaan ainoaan käsittelyyn siinä valmistelevasti käsitelty ed. Kantolan ym. toiv.al. n:o 220, joka sisältää yllämainitun ehdotuksen.

Puhemies: Käsittelyn pohjana on laki- ja talousvaliokunnan mietintö n:o 19.

Keskustelu:

Ed. Kantola: Herra puhemies! Pidän erittäin valitettavana sitä, että laki- ja talous-

(a) Source PDF transcript of plenary session 49/1967, p. 885.

```
Eduskunta yhtyy valiokunnan hylkäävään
ehdotukseen.

— Asia on loppuun käsitelty.

10) Fhdotus toivomukseksi määrärahasta lainoiksi
Uudenmaan läänin kunnille koulu-, sairaala-, asunto-
ja kunnallisteknillisten laitosten takentamiseksi.

Esitellään laki- ja talousvaliokunnan mie-
tintö n:o 19 ja otetaan ainoaan käsit-
telyyn siinä valmistelevasti käsitelty ed.
Kantolan ym. toiv.al, n:o 220, joka sisältää
yllämainitun ehdotuksen.

Puhemies: Käsittelyn pohjana on laki.
ja talousvaliokunnan mietintö n:o 19.

Keskustelu:

Ed. Kantola: Herra puhemies! Pidän
erittäin valitettavana sitä, että laki- ja talous-
```

(b) Result text after OCR.

Figure 3 Example of the source and the result after the OCR process.

metadata. The third file consists of the parliamentary session's plenary sessions and minutes transcripts. In the forming of URIs for the people and parties we once again utilized the PoF Ontology to ensure fluent linkage between the speech and prosopographical data sets.

6 Validation

The whole process extracted over 900 000 individual speeches from the whole period, from 1907 to current day. The length of a speech can vary from a single word to over thousand words in length. A completely automated process handling this much data is naturally prone to errors in dealing with exceptions in the data. At this point most validation of the result data has been manual. Currently, we are looking more deeply into the OCR results to get more concrete understanding of our success in that step of the process. Fig. 3 shows a snippet of the data in the original PDF format used and in the final text form. Apart from issues described in Section 5, Transformation Process, we have observed that the quality of the OCR results vary from decade to decade. The quality of 1990's OCR is quite good, with very little issues on relevant parts while results from the start of the 20^{th} century contain more errors. The main reason for these differences is the varying quality of available images and the paper the original document was printed on. A similar trend has been observed in [14].

Preliminary tests on speaker recognition (i.e., that each speech has speaker property value with speaker name and other required speaker metadata associated with it) show that after corrections the amount of recognized speakers tends to be over 99%. These tests were performed on random parliamentary sessions from all OCR-based decades. It is good to note that these numbers do not indicate whether the speaker is the correct one, as in some cases the chance of incorrect name correction or split speech does remain.

The RDF data model of the parliamentary debates is presented in a machine-processable format using the ShEx Shape Expressions language[31] [26]. We have made initial validation

[31] `https://shex.io`

experiments with PyShEx[32] and shex.js[33] validators. Based on the experiments, we have identified errors both in the schema and the data. The schema errors include syntax errors, incorrect cardinality definitions, incorrect literal datatype definitions, and incorrect namespaces for IRI values. The errors in the schema have been fixed accordingly. In the data, we have found systematic issues stemming from the RDF conversion process, e.g., some separate speeches and interruptions that were merged into one speech/interruption instance, speeches that were attached to multiple session item and diary (should be only one), and triples with an incorrectly minted object IRI (the base IRI of the Turtle file) instead of omitting the value altogether. The issues have been fixed in the data conversion process. We plan a full-scale ShEx validation phase integrated in the data conversion and publication process to spot and report errors in the dataset.

7 Publishing and Using Speeches via a Linked Open Data Service

The S-KG has been published on the Linked Data Finland platform[34] [8] according to the Linked Data publishing principles and other best practices of W3C [4], including, e.g., content negotiation and provision of a SPARQL[35] endpoint[36].

The data will be used via the SPARQL endpoint in two ways. Firstly, a portal called *ParliamentSampo – Finnish Parliament on the Semantic Web* is under development, a new member in the Sampo series of semantic portals[37]. The portal includes data analytic tools studying parliamentary debates, networks of Finnish politicians, and political culture, and is targeted to both researchers and the public for. Secondly, in addition to the ready-to-use application perspectives in the ParliamentSampo portal, the underlying SPARQL endpoint can and is being applied to custom data analyses in Digital Humanities research using YASGUI[38] [22] and Python scripting in Google Colab[39] and Jupyter[40] notebooks. In our work, the "FAIR guiding principles for scientific data management and stewardship" of publishing Findable, Accessible, Interoperable, and Re-usable data are used[41].

One example of using the data for analysis through SPARQL endpoint is shown in Fig. 4. It represents the number of speeches on a timeline by gender. The histogram shows the speeches of male speakers with a blue bar and female speakers with an orange bar. The green bar is for speeches where the speaker has not been identified due to speaker recognition issues described earlier. The chairpersons have been filtered out as they are often mentioned by the title in the data and therefore cannot be linked based on the speaker data to the actor data. With this in mind, it can be seen from the plot that the number of female speeches rises with time.

[32] https://github.com/hsolbrig/PyShEx
[33] https://github.com/shexSpec/shex.js
[34] https://ldf.fi
[35] https://www.w3.org/TR/sparql11-query/
[36] Access to this and the Parla-CLARIN dataset is currently restricted to consortium members.
[37] https://seco.cs.aalto.fi/applications/sampo/
[38] https://yasgui.triply.cc
[39] https://colab.research.google.com/notebooks/intro.ipynb
[40] https://jupyter.org
[41] https://www.go-fair.org/fair-principles/

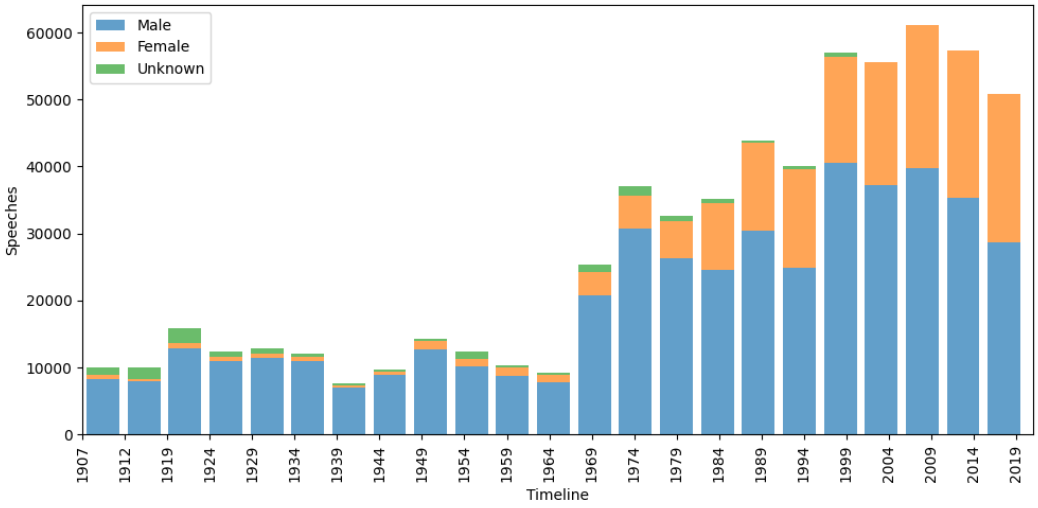


Figure 4 Total number of speeches by gender.

8 Discussion & Conclusions

This paper presented the first homogeneous publication of the full set of plenary speeches of PoF (1907–present) as a knowledge graph (S-KG) as Linked Data and in the emerging Parla-CLARIN standard. Thus far the speeches have been available only in PDF form, as text, in HTML, or in XML form depending on the time period and data publication.

Unlike in many other similar projects we have not focused only on a slice of existing data. Instead we have covered and brought into an unified format the speeches from the whole of Parliament of Finland's history. This makes it possible for any research to easily cover all of history with a single query and brings about completely new possibilities for further data analysis and research.

The main technical novelties in our approach w.r.t the related works discussed in Section 2 include the combined model of Parla-CLARIN and RDF developed for representing the speeches, integration of the data to the larger PoF Ontology for deeper data analyses, and enriching the data with a variety of external related national data sources to earn the 5th star according to the Linked Data 5-star model[42].

The variety of the pre-existing source formats is a key motivator for our work but also naturally a challenge. Bringing about a harmonious dataset from different sources is not a simple matter and requires familiarity with the source data. To deepen our understanding, we have also reached out to the Parliament's Central Office staff who are responsible for creating the minutes. This co-operation has been very beneficial.

The data has been published on the Linked Data Finland platform and is being used in Digital Humanities Research for studying the parliamentary language and political culture in the SEMPARL project and for implementing the end user applications. To earn the 6th star in Linked Data Finland model extending the 5-star model for better re-usability, the schema has been included and documented as part of the data publication, and to some

[42] https://www.w3.org/community/webize/2014/01/17/what-is-5-star-linked-data/

extent validated for the 7th star. The Parla-CLARIN data set has also been already taken into internal use in the consortium and while still undergoing revision, both data sets have proved promising and fit for use. The data and data service will be used also in the Helsinki Digital Humanities Hackathon[43] in May 2021 for feedback from external users. FinnParla data will eventually be opened during the SEMPARL project by the open license CC BY 4.0.

The S-KG data will be used as a basis of the semantic portal *ParliamentSampo – Finnish Parliament on the Semantic Web* that is being developed in the Semantic Parliament project, based on the Sampo model [5] and Sampo-UI framework [9]. The Parla-CLARIN version will also be made available to the public.

Regarding data enrichment, improvements in the keyword extraction mechanism as well as automatic recognition of broad topics in the dataset are planned for the near future. We also aim to further the combination of both presented formats by creating a third version of the data as LOD using Parla-CLARIN markup for the speech contents.

References

1 Kaspar Beelen, Timothy Alberdingk Thijm, Christopher Cochrane, Kees Halvemaan, Graeme Hirst, Michael Kimmins, Sander Lijbrink, Maarten Marx, Nona Naderi, Ludovic Rheault, and et al. Digitization of the canadian parliamentary debates. *Canadian Journal of Political Science*, 50(3):849–864, 2017. doi:10.1017/S0008423916001165.

2 Uldis Bojārs, Roberts Darģis, Uldis Lavrinovičs, and Pēteris Paikens. LinkedSaeima: A linked open dataset of Latvia's parliamentary debates. In Maribel Acosta, Philippe Cudré-Mauroux, Maria Maleshkova, Tassilo Pellegrini, Harald Sack, and York Sure-Vetter, editors, *Semantic Systems. The Power of AI and Knowledge Graphs*, pages 50–56, Cham, 2019. Springer–Verlag.

3 Eduskunta. Eduskunnan täysistunnot, ladattava versio 1.5, 2017. URL: http://urn.fi/urn:nbn:fi:lb-2019101721.

4 Tom Heath and Christian Bizer. *Linked Data: Evolving the Web into a Global Data Space (1st edition)*. Morgan & Claypool, Palo Alto, California, 2011. URL: http://linkeddatabook.com/editions/1.0/.

5 Eero Hyvönen. "Sampo" model and semantic portals for digital humanities on the semantic web. In *DHN 2020 Digital Humanities in the Nordic Countries. Proceedings of the Digital Humanities in the Nordic Countries 5th Conference*, pages 373–378. CEUR Workshop Proceedings, vol. 2612, October 2020. URL: http://ceur-ws.org/Vol-2612/poster1.pdf.

6 Eero Hyvönen, Petri Leskinen, Minna Tamper, Heikki Rantala, Esko Ikkala, Jouni Tuominen, and Kirsi Keravuori. Biographysampo – publishing and enriching biographies on the semantic web for digital humanities research. In Pascal Hitzler, Miriam Fernández, Krzysztof Janowicz, Amrapali Zaveri, Alasdair J.G. Gray, Vanessa Lopez, Armin Haller, and Karl Hammar, editors, *The Semantic Web. ESWC 2019*, pages 574–589. Springer-Verlag, June 2019. doi:10.1007/978-3-030-21348-0_37.

7 Eero Hyvönen, Minna Tamper, Arttu Oksanen, Esko Ikkala, Sami Sarsa, Jouni Tuominen, and Aki Hietanen. LawSampo: A semantic portal on a linked open data service for finnish legislation and case law. In *The Semantic Web: ESWC 2020 Satellite Events. Revised Selected Papers*, pages 110–114. Springer–Verlag, 2019.

8 Eero Hyvönen, Jouni Tuominen, Miika Alonen, and Eetu Mäkelä. Linked Data Finland: A 7-star model and platform for publishing and re-using linked datasets. In *ESWC 2014 Satellite Events*, pages 226–230. Springer-Verlag, 2014.

[43] http://heldig.fi/dhh21

9 Esko Ikkala, Eero Hyvönen, Heikki Rantala, and Mikko Koho. Sampo-UI: A Full Stack JavaScript Framework for Developing Semantic Portal User Interfaces. *Semantic Web – Interoperability, Usability, Applicability*, 2021. accepted.

10 Kimmo Kettunen and Matti La Mela. Digging deeper into the finnish parliamentary protocols – using a lexical semantic tagger for studying meaning change of everyman's rights (allemansrätten). In *DHN 2020 Digital Humanities in the Nordic Countries. Proceedings of the Digital Humanities in the Nordic Countries 5th Conference*, pages 63–80. CEUR Workshop Proceedings, vol. 2612, October 2020. URL: http://ceur-ws.org/Vol-2612/paper5.pdf.

11 Martijn Kleppe, Laura Hollink, Max Kemman, Damir Juric, Henri Beunders, Jaap Blom, Johan Oomen, and Geert-Jan Houben. Polimedia: Analysing media coverage of political debates by automatically generated links to radio & newspaper items. In *OKCon 2013 LinkedUp Veni Competition on Linked and Open Data for Education*, pages 63–80. CEUR Workshop Proceedings, vol. 1124, September 2013. URL: http://ceur-ws.org/Vol-1124/linkedup_veni2013_04.pdf.

12 Philipp Koehn. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86, 2005. URL: https://homepages.inf.ed.ac.uk/pkoehn/publications/europarl-mtsummit05.pdf.

13 Mikko Koho, Lia Gasbarra, Jouni Tuominen, Heikki Rantala, Ilkka Jokipii, and Eero Hyvönen. AMMO Ontology of Finnish Historical Occupations. In *Proceedings of the First International Workshop on Open Data and Ontologies for Cultural Heritage (ODOCH'19)*, volume 2375, pages 91–96. CEUR Workshop Proceedings, June 2019. URL: http://ceur-ws.org/Vol-2375/.

14 Matti La Mela. Tracing the emergence of nordic allemansrätten through digitised parliamentary sources. In Mats Fridlund, Mila Oiva, and Petri Paju, editors, *Digital histories: Emergent approaches within the new digital history*, pages 181–197. Helsinki University Press, 2020. doi:10.33134/HUP-5-11.

15 Emanuele Lapponi, Martin G. Søyland, Erik Velldal, and Stephan Oepen. The Talk of Norway: a richly annotated corpus of the Norwegian parliament, 1998–2016. *Language Resources and Evaluation*, 52(3):873–893, 2018. doi:10.1007/s10579-018-9411-5.

16 Petri Leskinen, Jouni Tuominen, and Eero Hyvönen. Members of parliament in finland (1907–) knowledge graph and its linked open data service, 2021. Submitted for review.

17 Eetu Mäkelä. Combining a REST lexical analysis web service with SPARQL for mashup semantic annotation from text. In *Proceedings of the ESWC 2014 demonstration track*, pages 424–428. Springer-Verlag, 2014. doi:10.1007/978-3-319-11955-7_60.

18 Arttu Oksanen, Jouni Tuominen, Eetu Mäkelä, Minna Tamper, Aki Hietanen, and Eero Hyvönen. Semantic Finlex: Transforming, publishing, and using Finnish legislation and case law as linked open data on the web. In G. Peruginelli and S. Faro, editors, *Knowledge of the Law in the Big Data Age*, volume 317 of *Frontiers in Artificial Intelligence and Applications*, pages 212–228. IOS Press, 2019.

19 Andrej Pancur and Tomaž Erjavec. The siParl corpus of Slovene parliamentary proceedings. In *Proceedings of the Second ParlaCLARIN Workshop*, pages 28–34, Marseille, France, 2020. European Language Resources Association. URL: https://www.aclweb.org/anthology/2020.parlaclarin-1.6.

20 Onni Pekonen. *Debating "the ABCs of parliamentary life": the learning of parliamentary rules and practices in the late nineteenth-century Finnish Diet and the early Eduskunta*. PhD thesis, University of Jyväskylä, Jyväskylä, 2014. URL: http://urn.fi/URN:ISBN:978-951-39-5843-5.

21 Christian Rauh, Pieter De Wilde, and Jan Schwalbach. The ParlSpeech data set: Annotated full-text vectors of 3.9 million plenary speeches in the key legislative chambers of seven European states, 2017. doi:10.7910/DVN/E4RSP9.

22 Laurens Rietveld and Rinke Hoekstra. The YASGUI family of SPARQL clients. *Semantic Web*, 8(3):373–383, 2017.

23 Katri Seppälä and Eero Hyvönen. Asiasanaston muuttaminen ontologiaksi. Yleinen suomalainen ontologia esimerkkinä FinnONTO-hankkeen mallista (Changing a keyword thesaurus into an ontology. General Finnish Ontology as an example of the FinnONTO model). Technical report, National Library, Plans, Reports, Guides, March 2014. URL: `https://www.doria.fi/handle/10024/96825`.

24 Osma Suominen. Annif: DIY automated subject indexing using multiple algorithms. *LIBER Quarterly*, 29(1):1–25, 2019. `doi:10.18352/lq.10285`.

25 Minna Tamper, Arttu Oksanen, Jouni Tuominen, Aki Hietanen, and Eero Hyvönen. Automatic annotation service APPI: Named entity linking in legal domain. In *Proceedings of ESWC 2020, Posters and Demos*. Springer–Verlag, 2020.

26 Katherine Thornton, Harold Solbrig, Gregory S. Stupp, Jose Emilio Labra Gayo, Daniel Mietchen, Eric Prud'hommeaux, and Andra Waagmeester. Using shape expressions (ShEx) to share RDF data models and to guide curation with rigorous validation. In Pascal Hitzler, Miriam Fernández, Krzysztof Janowicz, Amrapali Zaveri, Alasdair J.G. Gray, Vanessa Lopez, Armin Haller, and Karl Hammar, editors, *The Semantic Web. ESWC 2019*, pages 606–620. Springer-Verlag, 2019. `doi:10.1007/978-3-030-21348-0_39`.

27 Jouni Tuominen, Eero Hyvönen, and Petri Leskinen. Bio CRM: A data model for representing biographical data for prosopographical research. In *Biographical Data in a Digital World (BD2017)*, 2017. `doi:10.5281/zenodo.1040712`.

28 Astrid van Aggelen, Laura Hollink, Max Kemman, Martijn Kleppe, and Henri Beunders. The debates of the European Parliament as Linked Open Data. *Semantic Web*, 8(2):271–281, 2017. `doi:10.3233/SW-160227`.

29 Eero Voutilainen. Tekstilajitietoista kielenhuoltoa: puheen esittäminen kirjoitettuna eduskunnan täysistuntopöytäkirjoissa. In Liisa Tiittula and Pirkko Nuolijärvi, editors, *Puheesta tekstiksi – Puheen kirjallisen esittämisen alueita, keinoja ja rajoja*, pages 162–191. Suomalaisen Kirjallisuuden Seura, 2016.

Towards a Corpus of Historical German Plays with Emotion Annotations

Thomas Schmidt
Media Informatics Group, University of Regensburg, Germany

Katrin Dennerlein
German Literary Studies and Computational Literary Studies, University of Würzburg, Germany

Christian Wolff
Media Informatics Group, University of Regensburg, Germany

Abstract

In this paper, we present first work-in-progress annotation results of a project investigating computational methods of emotion analysis for historical German plays around 1800. We report on the development of an annotation scheme focussing on the annotation of emotions that are important from a literary studies perspective for this time span as well as on the annotation process we have developed. We annotate emotions expressed or attributed by characters of the plays in the written texts. The scheme consists of 13 hierarchically structured emotion concepts as well as the source (who experiences or attributes the emotion) and target (who or what is the emotion directed towards). We have conducted the annotation of five example plays of our corpus with two annotators per play and report on annotation distributions and agreement statistics. We were able to collect over 6,500 emotion annotations and identified a fair agreement for most concepts around a κ-value of 0.4. We discuss how we plan to improve annotator consistency and continue our work. The results also have implications for similar projects in the context of Digital Humanities.

2012 ACM Subject Classification Applied computing → Arts and humanities; Computing methodologies → Machine learning

Keywords and phrases Emotion, Annotation, Digital Humanities, Computational Literary Studies, German Drama, Sentiment Analysis, Emotion Analysis, Corpus

Digital Object Identifier 10.4230/OASIcs.LDK.2021.9

Funding This research is part of the project "Emotions in Drama" (Emotionen im Drama), is funded by the German Research Foundation (DFG) and part of the priority programme SPP 2207 Computational Literary Studies (CLS).

Acknowledgements We want to thank the following student annotators for their contributions to this project: Viola Hipler, Julia Jäger, Emma Ruß, and Leon Sautter.

1 Introduction

Emotions in dramatic texts are central for the dramaturgy, the characterization of characters, the intended effect on the reader as well as for the propagation of anthropological ideas. Emotions are a frequent and important subject in German literary studies of the 17th and 18th century. For example, literary scholars investigated the intended emotional effect [18, 39] or single emotions in plays of that time [2, 37]. We want to expand this hermeneutical research focused mostly on canonical texts. That is why we are applying computational emotion analysis on larger data sets of historical German plays around 1800. We are aiming at a more holistic view of emotion usage, progression and distribution in the plays of that time.

Computational emotion prediction in Natural Language Processing (NLP) describes the task of predicting the expressed emotion, predominantly in written text. Sentiment analysis, its neighbouring field, is focused on the prediction of the valence/polarity of text (if a text unit is rather positive or negative) while emotion prediction deals with more complex

emotion categories like anger, joy, or surprise [16]. Both methods have gained a lot of interest in Digital Humanities (DH) and Computational Literary Studies (CLS) (cf. [9]) and are applied to analyze emotions and sentiment in historical plays [12, 17, 23, 25, 26, 27, 29, 40], novels [6, 12, 21], fairy tales [1, 12], political texts [38], or online forums [14, 35]. DH projects also explore more modern literary genres like fan fictions [8, 7], original creative works on the web [19], subtitles of movies [5, 42] or song lyrics [24]. From a methodological point of view, many of these projects employ lexicon and rule-based methods to perform the sentiment and emotion analysis [1, 12, 17, 19, 23, 24, 25, 38, 40] leading to the development of lexicon-based sentiment analysis tools specifically designed for the DH-community [33]. However, these methods are outperformed by modern machine learning approaches [16]. The reason for the application of lexicon-based methods is the lack of well-annotated corpora of the particular domains that are necessary to train machine learning algorithms [31]. Currently, however, many projects work towards closing this gap and create first corpora of emotion annotated literary texts to explore deep learning based emotion analysis [7, 8]. Annotation of emotions and sentiments can be a challenging task [13, 41]. The task has been shown as even more problematic for historical and poetic texts [1, 28, 30, 36, 32, 38]. While the application of large-scale crowd-sourcing is common for many text types in NLP (cf. [13]), researchers rather refer to expert-based annotation for historical and poetic texts because of the challenges in language and interpretation [1, 28, 30, 36, 32, 38]. Furthermore, due to the high level of subjectivity and complexity of these texts, agreement statistics among expert or common annotators are oftentimes rather low [1, 28, 30, 36, 32, 38] which poses challenges to creating a valid gold standard. Recent research explores the development of tools with gamification elements to improve upon these problems [34, 42].

We present first results of a collaborative project between computer scientists and literary scholars exploring computational emotion analysis on German plays around 1800. Our main corpus currently consists of over 200 plays of that time and we performed our first annotation study on five representative plays of this corpus. We report on annotation results and how we address the challenges of emotion annotation in this field. We developed annotation schemes and processes that are more directed towards the literary scholar's perspective and goals than previous annotation schemes in NLP. Our experience with the annotation and overall results have implications for similar projects designing annotation schemes and performing emotion annotation in the context of CLS.

2 Annotation

In the following, we present the annotation scheme and process we developed. Please note that both the process as well as the scheme have been developed in an iterative process (cf. [22]) of pilot annotations on various scenes and plays of our corpus.

2.1 Annotation Scheme

We define emotion as a generic term for a character's state of mind of distinguishable quality at a given time that is expressed, among other channels, through written language. We annotate emotions experienced by the characters and attributed to them as they are represented as text. Please note that we are interested in the "real" intention and meaning of the expressions of the characters in the context of the entire play. For example, in the case of an ironic expression we annotate the intention of the character in this specific context and not what the text would mean independent of the content and context of the play.

We started the annotation scheme with a list of categorical emotions collected from various established systems of psychology (e.g. [20]) which is rather common in emotion prediction in NLP (cf. [13, 16]). However, we realized that these emotion concepts are missing core emotion and affect ideas important to capture the concept of "emotion" in plays of that time. These are important for the research of literary scholars, however. Therefore, we deviate from established psychological concepts of emotion and integrate concepts such as love and friendship which are not regarded as emotions in many psychological definitions (cf. [15]) but important for this type of literature. We continued with some pilot annotations with a very large scheme containing various emotional concepts important throughout literary history. However, the sheer size and complexity hindered the annotation process. The set was filtered on the most important concepts for literary studies for this time and genre. The final annotation scheme for emotions consists of the 13 concepts mentioned below. In brackets we include the German original terms since we do annotate in German. We translated them to English to the best of our knowledge, but semantic details might get skewed.

- Emotions of affection (*Emotionen der Zuneigung*)
  - Desire (*Lust*) (−)
  - Love (*Liebe*) (+)
  - Friendship (*Freundschaft*) (+)
  - Adoration (*Verehrung*) (+)
- Emotions of joy (*Emotionen der Freude*)
  - Joy (*Freude*) (+)
  - Schadenfreude (+)
- Emotions of fear (*Emotionen der Furcht*)
  - Fear (*Angst*) (−)
  - Despair (*Verzweiflung*) (−)
- Emotions of suffering (*Emotionen des Leids*)
  - Suffering (*Leid*) (−)
  - Compassion (*Mitleid*) (−)
  - Anger (*Ärger*) (−)
- Other
  - Hate (*Abscheu*) (−)
  - Emotional movement (*Emotionale Bewegtheit*)

We defined the set in a hierarchical order to deal with the imbalance problem or too few annotations in the later computational emotion prediction by mapping the emotions to the four main classes and two special types (*hate*, *emotional movement*). Emotional movement is used to annotate unspecific emotional arousal (that cannot be described with the other concepts) as well as astonishment. In the highest hierarchical order, emotions are represented by the two classes *positive* and *negative* (valence). We established a default valence for each emotion concept (marked as + and − in the above list) but annotators can also choose to deviate from this or mark an emotion as mixed via an attribute attached with each emotion annotation. *Schadenfreude*, although ambivalent, is assigned as positive per default since most of the time the emotion is perceived as positive by the experiencer in our texts. As with all emotions, annotators can however deviate from this assignment.

While this set of emotions is still not sufficient to fully capture emotional representations in the literature of that time, it is a compromise between the larger interest of literary scholars and the pragmatic limitations for the computational perspective as well as for the annotation process. Annotators annotate speeches (single utterances of a character separated

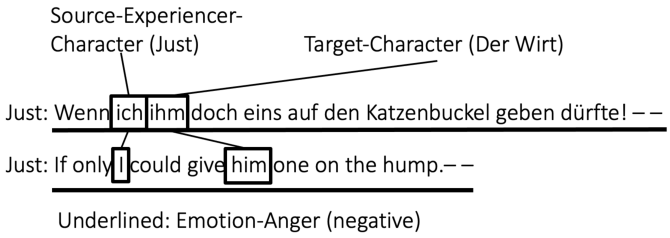


Figure 1 Illustration of an example annotation from Lessing's *Minna von Barnhelm* (Act 1, Scene 3): First line is the German original, second line an English translation. The entire sentence is annotated with anger. "Ich" (I) is the source, "ihm" (him) the target of the emotion. "Just" and "Der Wirt" are the names of the specific characters.

by the next utterance) and stage directions of the plays. They can annotate as much or little text as necessary but not spanning multiple speeches. Therefore, annotators can annotate single words, parts of sentences or multiple sentences. Text units can also consist of multiple or partially overlapping emotion annotations. We have decided to employ this variable and free annotation process since it is in line with the usual annotation work of literary scholars.

Following ideas of aspect-based sentiment analysis [7] we also annotate the source (the character experiencing or attributing an annotated emotion) and the target (the instance an emotion is directed towards). Similar to the emotion set, the set for source and target was adjusted and developed throughout multiple pilot annotation iterations and deemed important for the literary studies perspective since the emotional interaction of the characters are the main aspects of these plays. Source and target consist of the following sub types and possible attributes:

- Source
  - Experiencer: characters of the play, the author, impersonal, unknown
  - Attributing instance: characters of the play, the author, impersonal, unknown
- Target
  - Character: characters of the play, impersonal, unknown
  - Non-Character: animal, state, event or object

Impersonal is a mark for addressing the general public while *unknown* points to characters that are not in the original character list of the play, which is the standard selection of which the annotators can select the characters of the play by their name. The annotation window is as variable as with the emotion annotation. Annotators mark each explicit mention of source and target in the annotated text. In certain cases, it is however possible that an emotion annotation consists of neither source nor target. Figure 1 illustrates the annotation of one example speech of our corpus consisting of an emotion annotation and an explicit annotation of source and target of this annotation.

2.2 Corpus

To start the annotation we decided to annotate five plays of different genres and authors of our main corpus. Plays are annotated in their entirety since we are interested in context and content dependent annotations that need thorough interpretation of the entire plot. While this poses challenges to later generalization processes on the computational side, this is in line with the focus of this project on literary criticism. One aspect to deal with this problem is to annotate plays that are representative concerning content and language of

clusters of the 200 plays corpus. Most plays are taken from the *GerDracor* corpus [3], one play was taken from a free repository.[1] The following five plays have been annotated: *Minna von Barnhelm* (1767) by Lessing (comedy), *Kasperl' der Mandolettikrämer* (1789) by Eberl (comedy), *Kabale und Liebe* (1784) by Schiller (tragedy), *Menschenhass und Reue* (1790) by Kotzebue (comedy), *Faust. Eine Tragödie* (1807) by Goethe (tragedy).

2.3 Annotation Process

Since the annotation of the plays is dependent on deeper knowledge of the language and the content of the plays (as we perform context-aware annotation), crowd-sourcing annotations was not a viable option. In similar projects, annotations are performed by experts and semi-experts with a specific training [1, 28, 30, 31, 38]. In our setting, each play was annotated independently from each other by two students of German literary studies who are compensated monetarily for the annotations and who are employed in the research project. For the annotation of this corpus, we employed three annotators; each play was annotated in different combinations of annotator pairs. The students were introduced to the annotation guidelines by a literary scholar during multiple annotation training sessions and they were offered support during the annotation process. The students participated in the pilot annotation studies to determine the annotation scheme, as well. They had access to an annotation guidelines document consisting of a description of the scheme and multiple examples. The annotation was performed with the tool CATMA [4] for which we created the annotation scheme as described. The annotators were assigned to a play and had a specific deadline to finish it. Depending on the length of the play, each annotator had one to two weeks time to finish the annotation. On average the entire annotation process was performed throughout multiple days of the set time frame and took around 8–12 hours concerning the absolute duration.

3 Annotation Results

We collected over 6,500 emotion annotations for the five plays. First, we look at annotation distributions among the main and sub categories as well as statistics of the annotation lengths via token statistics (see Table 1).

The most frequent annotated emotions are *suffering* (15%) *joy* (13%), *anger* (13%) and *love* (12%). Some emotions are annotated rather rarely in our study e.g. *desire* (1%), *friendship* (2%) and *Schadenfreude* (3%). The main categories themselves are annotated more equally, however with a dominance of more negative emotion categories like the *emotions of suffering* (33%). *Emotional movement* has been proven as an important annotation category (12%). Looking at the size of annotations, all categories have rather similar averages (around 25 tokens) with a large variance ranging from one word annotations to larger paragraphs consisting of over 300 tokens. The dominance of negative emotions is supported by the distribution of the highest hierarchical order valence: 54% of all emotion annotations were either per default *negative* or marked via an attribute as *negative* compared to 34% of *positive* assignments. The remaining annotations in the highest hierarchical order were *emotional movement* annotations (11%). The possibility to select *mixed* as attribute for annotations was rarely used. This attribute has been shown to be rather redundant in our scheme since annotators can assign multiple emotions with differing valence to one text unit.

[1] `http://lithes.uni-graz.at/maezene/eberl_mandolettikraemer.html`

Table 1 Distribution of emotions and corresponding main categories. First, the sub emotions are listed followed by the summed results of the main categories in bold. Percentages are rounded.

Emotion	absolute	%	avg. tokens	min tokens	max tokens	std. tokens
Desire	50	1	23.22	4	83	16.49
Love	783	12	26.16	1	326	33.67
Friendship	127	2	22	1	120	18.66
Adoration	306	5	19.63	1	96	16.36
Emotions of affection	1,266	19	24.05	1	326	28.61
Joy	850	13	22.78	1	223	24.3
Schadenfreude	201	3	25.02	1	121	21.89
Emotions of joy	1,051	16	23.21	1	223	23.86
Fear	424	6	16.87	1	173	17.45
Despair	282	4	30.78	1	206	30.15
Emotions of fear	706	11	22.42	1	206	24.32
Suffering	998	15	26.12	1	302	28.91
Compassion	318	5	21.61	1	156	21.87
Anger	880	13	22.14	1	261	24.35
Emotions of suffering	2,196	33	23.87	1	302	26.27
Hate	614	9	25.05	1	167	26.19
Emotional movement	763	12	24.4	1	313	32.74

Table 2 Agreement statistics per play for the overall valence, the main emotion class and the sub emotions respectively for the text unit of speeches. κ refers to Cohen's κ while % is the proportion of agreed upon speeches among all speeches.

Drama	Valence (κ)	Valence (%)	Class (κ)	Class (%)	Emotion (κ)	Emotion (%)
Faust	0.44	67.853	0.345	59.399	0.342	58.064
Kabale und Liebe	0.382	58.908	0.325	50.313	0.312	47.992
Menschenhass und Reue	0.402	75.28	0.347	72.331	0.347	71.91
Minna von Barnhelm	0.406	74.619	0.377	72.752	0.356	71.23
Kasperl' der Mandolet-tikrämer	0.42	70.83	0.344	65.34	0.312	62.72
Overall	0.41	69.498	0.3476	64.027	0.333	62.383

Table 3 Source and target distributions. The sub categories are listed followed by the summed results of the main categories in bold. The percentages of the sub groups refer to the main class.

Annotation Type	absolute	%	avg. tokens	min tokens	max tokens	std. tokens
Experiencer	6,573	97	1.06	1	7	0.33
Attributing Instance	187	3	1.05	1	3	0.27
Source	6,760	50	1.06	1	7	0.33
Character	5,336	79	1.28	1	14	0.82
Non-Character	1,390	21	3.97	1	26	3.68
Target	6,726	50	1.84	1	26	2.13

Since the annotations are performed on variable text lengths, we decided on the following heuristic to calculate agreement among annotators: We focus on the speech and stage directions as central structural units of plays. They can consist of one word to multiple sentences. For every annotator we assign the specific emotion that is annotated the most (in total token count) for one speech. Thus, if multiple emotions are annotated, we assign the emotion that is annotated the most. We decided for this heuristic in order to be able to apply the traditional agreement metric Cohen's κ and get a first overview of agreement among annotators. We explore possibilities for more fitting fuzzy agreement metrics in future work. If no emotion was annotated the unit is marked as none. None is regarded as additional annotation class in this concept. Table 2 illustrates the agreements. The κ-value according to Cohen's κ is shown as well as the percentage wise agreement. We identified mostly moderate agreement for the valence according to [11] (0.41-0.6) and fair agreement for the main emotion category and the sub emotions (0.21-0.4). Due to the higher number of classes the agreement gets lower for the sub emotions.

We also gathered over 12,000 *source* and *target* annotations (see Table 3). Both classes are annotated to an equal extent. For sources, characters are mostly marked as *experiencer* of emotions (97%) and rarely as the ones attributing emotions to other characters (3%). Targets of emotions are mostly *characters* (79%). For the sub groups of theses classes, the following findings could be made. *Sources*, being it *experiencer* or *attributing instances*, are for the most part *one character* (94%) or *multiple characters* (2%). The attributes for *unknown* and *impersonal* sources are rarely used (2%). If a character is chosen as a *target*, the distribution is similar with *one to multiple characters* being the most frequent annotation (89%) compared to *unknown* (7%) and *impersonal* (4%). If the *target* is a *non-character*, the attribute assigned most frequently is *event* (61%) followed by *state* (19%), *objects* (16%) and *animals* (4%). Regarding the annotation lengths, source and target annotations are mostly one word annotations like pronouns or character names which points towards token based prediction mechanisms in later computational approaches to predict source and target.

4 Discussion

The annotated corpus will be made publicly available and is currently in the process of preparation.

To validate findings of the annotation analysis, we discussed our results with the annotators after the annotation. The extension of our scheme beyond established categories of psychology has been well received by annotators and we recommend this for similar projects. Concepts such as *love*, *suffering* and *emotional movement* are important parts of literature of that time and genre and have been annotated in large numbers. However, other concepts such as *desire* or *adoration* were rarely annotated. We are discussing the need for these concepts

since any complexity reduction of the scheme is beneficial for annotation speed, consistency and the later prediction. Please note that the annotation of emotions is highly influenced by the plays chosen to be annotated. Concepts such as *desire* and *adoration* are more important for earlier periods which we will investigate in the future and which will likely lead to the collection of more annotations. Looking at the main categories, the distribution becomes more equal. Negative categories are more frequent, although the majority of our chosen corpus consists of comedies. This is in line with previous annotation results in similar contexts [1, 28, 30] showing that negativity is an integral part of the narrative of most plays. The genre assignment comedy just points towards a positive ending, the play itself still consists of conflicts and disputes up until the end. Annotators rarely used the annotation of the attribute *mixed* for emotion. This attribute is redundant in our scheme and will be discarded. Considering *source* and *target*, we identified that annotators mostly annotate them as characters and not as non-characters which is quite intuitive in light of the content of the plays which are driven by emotional interactions of characters. We will reflect upon the question if differentiated sub classes for *non-characters* make sense if this main class is annotated rather rarely. The variable annotation lengths have also been perceived rather positively by the annotators and we also recommend the application of this idea for similar projects in a literary studies context. Emotions were annotated in variable sizes concerning number of words and sentences. This resembles the reality of the emotion expressions in these plays and is also in line with the general annotation behavior of literary scholars. Forcing annotations for a concrete window size would be challenging for decision processes during the annotation and would prolong and complicate the process. We plan to apply heuristics to map annotations on structural units and perform speech, sentence, n-gram and token based multi-label emotion prediction in our computational approaches.

The current agreement results indicate fair to moderate agreement. This is mostly in line with results of projects with similar text types [1, 28, 30, 32, 36, 38] since the material is more subjective and challenging to interpret. Our approach to perform context-sensitive annotation reinforces this aspect. In future work, we plan to explore sentence and token based agreements but also agreements of source and target annotations to get a better overview of the annotation problems. We also see potential in fuzzy agreement scores to represent the agreement in our variable and complex setting in a more fitting way [10] since our heuristic certainly leads to further disagreement in certain instances. Furthermore, we argue that we will reach higher agreements the more experience the annotators gain. To support this process and to find a way to deal with the disagreements among the annotators, we decided to add a subsequent post-annotation phase after the first two independent annotations by the students. This post-annotation phase is performed under the guidance of a literary scholar expert annotator who discusses the annotation with the students and creates a *consensus annotation* during these sessions. Although this might increase the annotation duration, it will improve the understanding of all annotators and might lead to more consistent annotations. Kajava et al. [5] argue that κ-values of 0.6 are acceptable for multi-label emotion annotations to validate the consistency of a scheme. The consensus annotation will also be the material we use to train and evaluate computational emotion analysis based on machine learning. We will adjust the annotation scheme and continue the annotations in the described way.

References

1 Cecilia Ovesdotter Alm and Richard Sproat. Emotional Sequencing and Development in Fairy Tales. In Jianhua Tao, Tieniu Tan, and Rosalind W. Picard, editors, *Affective Computing and Intelligent Interaction*, Lecture Notes in Computer Science, pages 668–674, Berlin, Heidelberg, 2005. Springer. doi:10.1007/11573548_86.

2 Thomas Anz. Todesszenarien : literarische Techniken zur Evokation von Angst, Trauer und anderen Gefühlen. In Lisanne Ebert, editor, *Emotionale Grenzgänge. Konzeptualisierungen von Liebe, Trauer und Angst in Sprache und Literatur*, pages 113–129. Königshausen & Neumann, Würzburg, 2011.

3 Frank Fischer, Ingo Börner, Mathias Göbel, Angelika Hechtl, Christopher Kittel, Carsten Milling, and Peer Trilcke. Programmable Corpora: Introducing DraCor, an Infrastructure for the Research on European Drama, July 2019. Conference Name: Digital Humanities 2019: "Complexities" (DH2019) Publisher: Zenodo. doi:10.5281/zenodo.4284002.

4 Evelyn Gius, Jan Christoph Meister, Marco Petris, Malte Meister, Christian Bruck, Janina Jacke, Mareike Schuhmacher, Marie Flüh, and Jan Horstmann. CATMA, 2020. doi:10.5281/zenodo.4353618.

5 Kaisla Kajava, Emily Öhman, Piao Hui, and Jörg Tiedemann. Emotion Preservation in Translation: Evaluating Datasets for Annotation Projection. In *Proceedings of Digital Humanities in Nordic Countries (DHN 2020)*, pages 38–50. CEUR, 2020.

6 Tuomo Kakkonen and Gordana Galić Kakkonen. SentiProfiler: Creating Comparable Visual Profiles of Sentimental Content in Texts. In *Proceedings of the Workshop on Language Technologies for Digital Humanities and Cultural Heritage*, pages 62–69, Hissar, Bulgaria, 2011. Association for Computational Linguistics.

7 Evgeny Kim and Roman Klinger. Who Feels What and Why? Annotation of a Literature Corpus with Semantic Roles of Emotions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1345–1359, Santa Fe, New Mexico, USA, 2018. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/C18-1114.

8 Evgeny Kim and Roman Klinger. An Analysis of Emotion Communication Channels in Fan-Fiction: Towards Emotional Storytelling. In *Proceedings of the Second Workshop on Storytelling*, pages 56–64, Florence, Italy, August 2019. Association for Computational Linguistics.

9 Evgeny Kim and Roman Klinger. A Survey on Sentiment and Emotion Analysis for Computational Literary Studies. *Zeitschrift für digitale Geisteswissenschaften*, 2019. arXiv: 1808.03137. doi:10.17175/2019_008.

10 Andrei P. Kirilenko and Svetlana Stepchenkova. Inter-Coder Agreement in One-to-Many Classification: Fuzzy Kappa. *PLOS ONE*, 11(3):e0149787, 2016. Publisher: Public Library of Science. doi:10.1371/journal.pone.0149787.

11 J. Richard Landis and Gary G. Koch. The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159–174, 1977. Publisher: [Wiley, International Biometric Society].

12 Saif Mohammad. From Once Upon a Time to Happily Ever After: Tracking Emotions in Novels and Fairy Tales. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 105–114, Portland, OR, USA, 2011. Association for Computational Linguistics.

13 Saif Mohammad. A Practical Guide to Sentiment Annotation: Challenges and Solutions. In *Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 174–179, San Diego, California, June 2016. Association for Computational Linguistics. doi:10.18653/v1/W16-0429.

14 Luis Moßburger, Felix Wende, Kay Brinkmann, and Thomas Schmidt. Exploring online depression forums via text mining: A comparison of Reddit and a curated online forum. In *Proceedings of the Fifth Social Media Mining for Health Applications Workshop & Shared Task*, pages 70–81, Barcelona, Spain (Online), December 2020. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/2020.smm4h-1.11.

15 Kevin Mulligan and Klaus R. Scherer. Toward a Working Definition of Emotion. *Emotion Review*, 4(4):345–357, 2012. Publisher: SAGE Publications. doi:10.1177/1754073912445818.

16 Mika V. Mäntylä, Daniel Graziotin, and Miikka Kuutila. The evolution of sentiment analysis – A review of research topics, venues, and top cited papers. *Computer Science Review*, 27:16–32, February 2018. doi:10.1016/j.cosrev.2017.10.002.

17 Eric T. Nalisnick and Henry S. Baird. Character-to-Character Sentiment Analysis in Shakespeare's Plays. In *Proceedings of the 51st Annual Meeting of the Association for Computa-*

tional Linguistics (Volume 2: Short Papers), pages 479–483, Sofia, Bulgaria, August 2013. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/P13-2085.

18 Winfried Nolting. *Die Dialektik der Empfindung: Lessings Trauerspiele "Miss Sara Sampson" und "Emilia Galotti": mit einer Einleitung, gemischte Gefühle: zur Problematik eines explikativen Verstehens des Empfindung.* Number 1 in Studien zu einer Geschichte der literarischen Empfindung / Winfried Nolting. F. Steiner Verlag Wiesbaden, Stuttgart, 1986.

19 Federico Pianzola, Simone Rebora, and Gerhard Lauer. Wattpad as a resource for literary studies. Quantitative and qualitative examples of the importance of digital social reading and readers' comments in the margins. *PLOS ONE*, 15(1):e0226708, 2020. Publisher: Public Library of Science. doi:10.1371/journal.pone.0226708.

20 Robert Plutchik. *Emotion, a Psychoevolutionary Synthesis.* Harper & Row, 1980. Google-Books-ID: G5t9AAAAMAAJ.

21 Andrew J. Reagan, Lewis Mitchell, Dilan Kiley, Christopher M. Danforth, and Peter Sheridan Dodds. The emotional arcs of stories are dominated by six basic shapes. *EPJ Data Science*, 5(1):31, 2016. arXiv: 1606.07772. doi:10.1140/epjds/s13688-016-0093-1.

22 Nils Reiter. Anleitung zur Erstellung von Annotationsrichtlinien. In *Reflektierte algorithmische Textanalyse*, pages 193–202. De Gruyter, 2020. doi:10.1515/9783110693973-009.

23 Thomas Schmidt. Distant reading sentiments and emotions in historic german plays. In *Abstract Booklet, DH_Budapest_2019*, pages 57–60. Budapest, Hungary, September 2019. URL: https://epub.uni-regensburg.de/43592/.

24 Thomas Schmidt, Marlene Bauer, Florian Habler, Hannes Heuberger, Florian Pilsl, and Christian Wolff. Der einsatz von distant reading auf einem korpus deutschsprachiger songtexte. In Christof Schöch, editor, *DHd 2020: Spielräume; Digital Humanities zwischen Modellierung und Interpretation. Konferenzabstracts; Universität Paderborn, 02. bis 06. März 2020*, pages 296–300. Paderborn, Germany, 2020. URL: https://epub.uni-regensburg.de/43704/.

25 Thomas Schmidt and Manuel Burghardt. An Evaluation of Lexicon-based Sentiment Analysis Techniques for the Plays of Gotthold Ephraim Lessing. In *Proceedings of the Second Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 139–149, Santa Fe, New Mexico, August 2018. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/W18-4516.

26 Thomas Schmidt and Manuel Burghardt. Toward a Tool for Sentiment Analysis for German Historic Plays. In Michael Piotrowski, editor, *COMHUM 2018: Book of Abstracts for the Workshop on Computational Methods in the Humanities 2018*, pages 46–48, Lausanne, Switzerland, June 2018. Laboratoire laussannois d'informatique et statistique textuelle. URL: https://zenodo.org/record/1312779.

27 Thomas Schmidt, Manuel Burghardt, and Katrin Dennerlein. "Kann man denn auch nicht lachend sehr ernsthaft sein?" – Zum Einsatz von Sentiment Analyse-Verfahren für die quantitative Untersuchung von Lessings Dramen. In Georg Vogeler, editor, *Kritik der digitalen Vernunft. Abstracts zur Jahrestagung des Verbandes Digital Humanities im deutschsprachigen Raum, 26.02. - 02.03.2018 an der Universität zu Köln*, pages 244–249, Köln, 2018. Universitäts- und Stadtbibliothek Köln.

28 Thomas Schmidt, Manuel Burghardt, and Katrin Dennerlein. Sentiment annotation of historic german plays: An empirical study on annotation behavior. In Sandra Kübler and Heike Zinsmeister, editors, *annDH 2018, Proceedings of the Workshop on Annotation in Digital Humanities 2018 (annDH 2018), Sofia, Bulgaria, August 6-10, 2018*, pages 47–52. RWTH Aachen, Aachen, August 2018. URL: https://epub.uni-regensburg.de/43701/.

29 Thomas Schmidt, Manuel Burghardt, Katrin Dennerlein, and Christian Wolff. Katharsis – a tool for computational drametrics. In *Book of Abstracts, Digital Humanities Conference 2019 (DH 2019)*. Utrecht, Netherlands, 2019. URL: https://epub.uni-regensburg.de/43579/.

30 Thomas Schmidt, Manuel Burghardt, Katrin Dennerlein, and Christian Wolff. Sentiment annotation for lessing's plays: Towards a language resource for sentiment analysis on german literary texts. In Thierry Declerck and John P. McCrae, editors, *2nd Conference on Language,*

Data and Knowledge (LDK 2019), pages 45–50. RWTH Aachen, Aachen, May 2019. URL: https://epub.uni-regensburg.de/43569/.

31 Thomas Schmidt, Manuel Burghardt, and Christian Wolff. Herausforderungen für Sentiment Analysis-Verfahren bei literarischen Texten. In Manuel Burghardt and Claudia Müller-Birn, editors, *INF-DH-2018*, Berlin, Germany, September 2018. Gesellschaft für Informatik e.V. doi:10.18420/infdh2018-16.

32 Thomas Schmidt, Manuel Burghardt, and Christian Wolff. Toward Multimodal Sentiment Analysis of Historic Plays: A Case Study with Text and Audio for Lessing's Emilia Galotti. In Costanza Navarretta, Manex Agirrezabal, and Bente Maegaard, editors, *Proceedings of the Digital Humanities in the Nordic Countries 4th Conference*, volume 2364 of *CEUR Workshop Proceedings*, pages 405–414, Copenhagen, Denmark, March 2019. CEUR-WS.org. URL: http://ceur-ws.org/Vol-2364/37_paper.pdf.

33 Thomas Schmidt, Johanna Dangel, and Christian Wolff. Senttext: A tool for lexicon-based sentiment analysis in digital humanities. In Thomas Schmidt and Christian Wolff, editors, *Information Science and its Neighbors from Data Science to Digital Humanities. Proceedings of the 16th International Symposium of Information Science (ISI 2021)*, volume 74, pages 156–172. Werner Hülsbusch, Glückstadt, 2021. URL: https://epub.uni-regensburg.de/44943/.

34 Thomas Schmidt, Marco Jakob, and Christian Wolff. Annotator-centered design: Towards a tool for sentiment and emotion annotation. In Claude Draude, Martin Lange, and Bernhard Sick, editors, *INFORMATIK 2019: 50 Jahre Gesellschaft für Informatik – Informatik für Gesellschaft (Workshop-Beiträge)*, pages 77–85, Bonn, 2019. Gesellschaft für Informatik e.V. doi:10.18420/inf2019_ws08.

35 Thomas Schmidt, Florian Kaindl, and Christian Wolff. Distant reading of religious online communities: A case study for three religious forums on reddit. In *DHN*, pages 157–172, Riga, Latvia, 2020.

36 Thomas Schmidt, Brigitte Winterl, Milena Maul, Alina Schark, Andrea Vlad, and Christian Wolff. Inter-rater agreement and usability: A comparative evaluation of annotation tools for sentiment annotation. In Claude Draude, Martin Lange, and Bernhard Sick, editors, *INFORMATIK 2019: 50 Jahre Gesellschaft für Informatik – Informatik für Gesellschaft (Workshop-Beiträge)*, pages 121–133, Bonn, 2019. Gesellschaft für Informatik e.V. doi:10.18420/inf2019_ws12.

37 Anja Schonlau. *Emotionen im Dramentext: eine methodische Grundlegung mit exemplarischer Analyse zu Neid und Intrige 1750-1800*. Number Band 25 in Deutsche Literatur. De Gruyter, Berlin Boston, 2017. OCLC: 978262308.

38 Rachele Sprugnoli, Sara Tonelli, Alessandro Marchetti, and Giovanni Moretti. Towards sentiment analysis for historical texts. *Digital Scholarship in the Humanities*, 31:762–772, 2015. Publisher: Oxford : Oxford University Press. doi:10.1093/llc/fqv027.

39 Hermann Wiegmann, editor. *Die ästhetische Leidenschaft: Texte zur Affektenlehre im 17. und 18. Jahrhundert*. Number 27 in Germanistische Texte und Studien. Olms, Hildesheim, 1987. OCLC: 15741918.

40 Mehmet Can Yavuz. Analyses of Character Emotions in Dramatic Works by Using EmoLex Unigrams. In *Proceedings of the Seventh Italian Conference on Computational Linguistics, CLiC-it'20*. Bologna, Italy, 2021.

41 Emily Öhman. Challenges in Annotation: Annotator Experiences from a Crowdsourced Emotion Annotation Task. In *Proceedings of the Digital Humanities in the Nordic Countries 5th Conference*, pages 293–301. CEUR Workshop Proceedings, 2020.

42 Emily Sofi Öhman and Kaisla S. A. Kajava. Sentimentator: Gamifying Fine-grained Sentiment Annotation. In *Digital Humanities in the Nordic Countries 2018*, pages 98–110. CEUR Workshop Proceedings, 2018.

Enriching a Lexical Resource for French Verbs with Aspectual Information

Anna Kupść
CLLE and Université Bordeaux Montaigne, France

Pauline Haas
UMR Lattice 8094 and Université Paris 13, France

Rafael Marín
UMR STL 8163, CNRS and Université de Lille, F-59000, France

Antonio Balvet
UMR STL 8163, CNRS and Université de Lille, F-59000, France

Abstract

The paper presents a syntactico-semantic lexicon of over a thousand French verbs. It has been created by manually adding lexical aspect features to verb frames from TreeLex [16]. We present how the original syntactic resource has been adapted to the current project, our aspect assignment procedure and an overview of the resulting lexical resource.

2012 ACM Subject Classification Computing methodologies → Language resources; Computing methodologies → Lexical semantics; Computing methodologies → Information extraction

Keywords and phrases computational semantics, corpora-based methods in language engineering, electronic language resources and tools, formalization of natural languages

Digital Object Identifier 10.4230/OASIcs.LDK.2021.10

Supplementary Material
Dataset: `http://redac.univ-tlse2.fr/lexiques/treelexPlusPlus.html`

1 Introduction

For Natural Language Processing (e.g., Information Extraction, Syntactic Parsing, Text Generation), as well as language-oriented Digital Humanities applications (e.g., Discourse Analysis, stylometry), machine-tractable as well as human-readable large-scale lexical resources are still a very valuable asset, even in a scene which appears today dominated by robust Machine-Learning algorithms and giga-word corpora. For instance, even though syntactic parsing has seen great advances in the past 10 years, thanks to the development of Treebanks and dependency-annotated corpora, even the best parser fails to capture in a consistent and predictable way such an intuitive linguistic notion as transitivity. In this sense, (semi-)manually constructed lexicons are an indispensable complementary resource to corpus-driven resources (e.g., "word embeddings", n-grams datasets). We see the symbolic/-Machine Learning divide as a consequence of the fact that each type of resource addresses a portion of the problem. Thus, the challenge contemporary NLP systems are facing today is more how to integrate different knowledge sources than to prove that one source is better – or more consistent – than the other. In this paper, we present TreeLex++, an extension of TreeLex [16], a syntactic lexicon for French, based on the French Treebank (FTB), enriched here with aspectual information. Different lexical resources have been devised over several decades for the automatic processing of French texts, in different theoretical frameworks: from the manually-encoded Lexicon-Grammar tables [13] framed in a distributionalist framework, to contemporary large-scale, semi-automatically induced lexicons such as the Lefff [24, 23], or resources acquired by way of "serious games", such as Jeux de Mots [17, 18]. Most of those

lexical resources have focused on providing a formalized description of the main syntactic categories, with an emphasis on verbal predicates. In extending TreeLex with aspectual information, our goal is primarily to set up a large-scale aspectual characterization process of verbs. Secondly, we wish to provide the NLP and DH communities with a resource which combines corpus-induced syntactic characterizations[1] as well as basic aspectual distinctions, based on Vendler's classification [25].

In the first sections, we present how TreeLex++ derives from the original FTB-induced TreeLex resource (Section 2 and 3). Then we move on to the presentation of our aspectual semantics characterization process (Section 4). In Section 5 we give a general overview of the present state of the resource. Section 6 is dedicated to conclusions and perspectives.

2 TreeLex

TreeLex is a syntactic lexicon automatically extracted from the French Treebank [1]. The lexicon contains ca. 2000 contemporary French verbs with their syntactic realizations and frequencies found in the FTB. The FTB is a corpus of newspaper texts (*Le Monde* newspaper, 1990–1993), in which constituent trees were originally encoded in XML format. In addition to lexical information for every word (category, lemma, person, number, gender etc.), the corpus provides a syntactic structure for each sentence: both syntactic groups and functions are indicated (see Figure 1).

```
<SENT argument="ETR" author="MINANGOY ROBERT" date="1990-01-19" nb="1006" textID="456">
  <NP fct="SUJ">
    <w cat="D" ee="D-def-ms" ei="Dms" lemma="le" mph="ms" subcat="def">Le</w>
    <w cat="A" ee="A-ord-ms" ei="Ams" lemma="deuxième" mph="ms" subcat="ord">deuxième</w>
    <w cat="N" ee="N-C-ms" ei="NCms" lemma="problème" mph="ms" subcat="C">problème</w>
  </NP>
  <VN>
    <w cat="V" ee="V--P3s" ei="VP3s" lemma="être" mph="P3s" subcat="">est</w>
  </VN>
  <NP fct="ATS">
    <w cat="D" ee="D-def-fs" ei="Dfs" lemma="le" mph="fs" subcat="def">la</w>
    <w cat="N" ee="N-C-fs" ei="NCfs" lemma="nourriture" mph="fs" subcat="C">nourriture</w>
  </NP>
  <w cat="PONCT" ee="PONCT-S" ei="PONCTS" lemma="." subcat="S">.</w>
</SENT>
```

Figure 1 A sample of FTB sentence annotation.

The XML-based annotation schema has since been complemented with a more straightforward tabulated format, following the CoNLL specifications that were widely adopted after the CoNLL shared task on dependency-parsing [20].

The FTB annotation schema is centered around the verbal nucleus (VN) which makes syntactic dependents easily accessible. This corpus organization is exploited by [16] in order to obtain obligatory arguments and provide syntactic frames for verbs present in the FTB. The resulting lexicon, called TreeLex[2], provides a rich syntactic representation of each argument since both functions and their phrasal realizations are encoded. Example 1 shows a lexical entry for the transitive verb *entraver* 'to impede' which takes a nominal subject (SUJ:NP) and a nominal direct object (OBJ:NP).

[1] As opposed to theory-driven ones.
[2] `http://redac.univ-tlse2.fr/lexiques/treelex_en.html`

Table 1 TreeLex functions with syntactic realizations.

Tag	Function	Possible phrasal realizations
SUJ	subject	NP, VPinf, Ssub
OBJ	direct object	NP, VPinf, Ssub
A-OBJ	indirect object introduced by *à*	VPinf, PP
DE-OBJ	indirect object introduced by *de*	VPinf, PP
P-OBJ	indirect prepositional object (other than *de* and *à*)	PP
ATS	subject complement	AP, NP, VPpart, VPinf, Ssub
ATO	direct object complement	AP, NP, VPpart, VPinf, Ssub
ref	obligatory reflexive clitic pronoun	CL
obj	other obligatory clitic pronoun	en, y

1. entraver: SUJ:NP,OBJ:NP

In Treelex, names of functions and syntactic constituents are adopted directly from the FTB notation, with two additions (`ref` and `obj`) for obligatory clitics, cf. Table 1. Arguments with clitic realizations are used to indicate reflexive verbs (ex., *se réjouir* 'to rejoice': SUJ:NP,ref:CL), idiomatic expressions (ex., *s'en sortir* 'to cope/get through': SUJ:NP,obj:en,ref:CL) or an impersonal subject (ex., *falloir* 'to have to': SUJ:il,OBJ:VPinf).

If a verb allows for different syntactic combinations (i.e., either a list of functions or different realizations), every frame is listed separately. Therefore, a single verb (more precisely, its lemma) can be found several times in the lexicon, see (2). As no semantic disambiguation was performed, this strategy aims at distinguishing potentially different senses associated with each frame. Here, in (2a-b), *voler* has the meaning of 'to steal' whereas in (2c) it can be translated as 'to fly'.

2. **(a)** *voler*: SUJ:NP,OBJ:NP,A-OBJ:NP
(b) *voler*: SUJ:NP,DE-OBJ:NP
(c) *voler*: SUJ:NP

As noted on TreeLex's website, an optional realization of specific arguments has been added manually, cf. (3).

3. *détruire*: SUJ:NP,(OBJ:NP)

Finally, since multi-word units are indicated in the FTB, TreeLex lists 465 multi-word verbs, such as *courir le risque* 'to take a risk' or *donner lieu* 'to result/take place'.

3 Beyond TreeLex: towards TreeLex++

TreeLex contains 1912 verbs and 3229 entries, i.e., verb-frame couples, which correspond to 24660 verb occurrences[3] attested in the FTB corpus. The resource provides a rich set of syntactic information and, as stated in [16, p.38], it can be easily integrated with other resources for NLP tasks such as parsing, or text generation. However, its relatively small size makes open-domain applications problematic.

[3] We present here figures from the on-line TreeLex version, `http://redac.univ-tlse2.fr/lexiques/treelex/treelex_verbs.csv`.

On the other hand, TreeLex's size makes an in-depth qualitative linguistic study feasible. For example, it could be extended with semantic information to investigate interactions between semantic and syntactic properties of verbs. For French, several projects have produced lexical resources containing syntactic and semantic verbal properties, or different levels of semantic information, e.g., verbal semantic classes (LVF, cf. [10]), thematic roles (French FrameNet, cf. [7]) or lexical aspect (Nomage, cf. [3] or [9]). In the current project, we decided to focus on high-level syntax-semantics relationships and thus we augmented the syntactic frames in TreeLex with manually encoded aspectual information. Our approach differs from [3] or [9], as verbal aspect assignment is guided by corpus examples rather than by elicited sentences.[4] Similarly to [9], aspect is assigned to a verb-frame couple rather than to a verb alone. Nevertheless, the level of detail of our aspectual classes is distinct both from [3] and [9]: we use only the four major Vendlerian classes[5] .

In order to prepare the TreeLex data for aspect assignment, several modifications have been adopted. First, all frames had to be represented in a uniform way. Therefore all syntactic arguments, whether optional or not, have been treated equally and indications of optional realizations have been removed. In particular, verbs such as *détruire* 'to destroy' in (3) were transformed into (4):

4. détruire: SUJ:NP,OBJ:NP

Second, we had to address the ambiguity in TreeLex entries. As shown in (2), TreeLex verbs may appear with several frames. According to [16], this affects about 40% of TreeLex verbs. Such multiple frames may indicate a polysemous and/or a polyaspectual verb. However, all different syntactic realizations of a single argument structure (the same sequence of functions) are listed as separate frames in TreeLex, see (5). This representation is therefore unclear: it may show a true semantic (meaning) difference or introduce an artificial syntactic (frame) ambiguity. For example, the direct object (OBJ) of the verb *déplorer* 'to regret/deplore' in (5) has two syntactic realizations (a nominal phrase, NP, or a subordinate phrase, Ssub) but this syntactic variation does not imply a difference in meaning.

5. (a) déplorer: SUJ:NP,OBJ:Ssub
(b) déplorer: SUJ:NP,OBJ:NP

In order to avoid such an artificial ambiguity, we grouped all frames which differed only by their phrasal realization. Therefore, the double nature of OBJ in (5) is currently represented as in (6).

6. déplorer: SUJ:NP,OBJ:NP/Ssub

In an effort to reduce semantic ambiguity, we decided to consider only verbs which, after syntactic grouping, appeared with a single syntactic frame. As a consequence, verbs such as *voler* in (2) have been excluded.[6] Multi-word verbal units have been omitted as well, as their meaning is usually idiosyncratic and conventional. Moreover, due to their idiomatic nature, syntactic construction appears heavily constrained.

Finally, all remaining 1161 verbs have been coupled with examples extracted from the FTB. We collected corpus examples in order to illustrate how each frame is instantiated and to provide a real context for aspect assignment.

---

[4] [3] use corpus examples to assign aspectual properties only to nouns. Verbs are annotated with no explicit contextual information.

[5] See Section 4 for details.

[6] This strategy does not replace a real semantic disambiguation since verbs which allow for a single syntactic frame may still be polysemous. This issue will be addressed in further sections.

Table 2 The four situation types, based on [25].

Class	Dynamic	Durative	Telic
STATE	−	+	−
ACT	+	+	−
ACC	+	+	+
ACH	+	−	+

4 Incorporating lexical aspect

Aspectual information has been added manually to TreeLex verbs. Unlike grammatical aspect, lexical aspect refers to inherent semantic properties indicating the way in which predicates are structured in relation to time. In the most general terms, the properties in question have to do with the presence (or lack thereof) of an end point (limit or boundary), duration or dynamicity in the lexical structure of certain classes of verbs. Thus, for instance, the presence of a limit distinguishes between **telic** (i.e., a time-limited situation) and **atelic** verbs.

These semantic properties give rise to four major aspectual classes (cf. [25]): STATE, ACTIVITY (ACT), ACCOMPLISHMENT (ACC) and ACHIEVEMENT (ACH). Their semantic features are listed in Table 2.

4.1 Annotation procedure

Aspectual assignment is a relatively new task, in the field of natural language annotation. The research exposed here is therefore to be seen as the first steps towards a full-fledged syntactic/semantic lexical resource. Our aspect assignment procedure consisted in a double manual annotation by two experts in semantics. Our annotation procedure is therefore not a "standard" annotation process, since, after the initial annotation phase, a final adjudication phase took place in order to arrive at the annotations presented in the current version of Treelex++. This process, which departs from established annotation approaches, is to be considered as a way of ensuring consistency in the current phase, where aspectual tagging is entirely performed manually. Each verb has been considered along with its syntactic frame and the corresponding examples found in the FTB. The assignment task consisted in choosing one of the four classes (tags) in Table 2. Each decision was made after applying the usual tests presented in the literature on verb lexical aspect (see [12, 15, 25, 8, 27, 19, 6, 22], among others). We have used the following six tests (cf. Table 3):

- **T1**: progressive form of *être en train de* 'to be V-ing'
- **T2**: question related to dynamicity *Que s'est-il passé hier?* 'What happened yesterday?'
- **T3**: use of aspectual semi-auxiliaries *commencer à* 'to start doing something', *continuer de* 'to keep on doing something', *arrêter de* 'to stop doing something'
- **T4**: duration complement *en x temps* 'in x time'
- **T5**: duration complement *pendant x temps* 'during x time'
- **T6**: imperfective paradox *V[temps inaccompli] IMPLIQUE V [temps accompli]* 'V[imperfect tense] IMPLIES V [perfect tense]'

Table 3 A grid for the allocation of aspectual classes to TreeLex verbs.

Situation type	T1	T2	T3	T4	T5	T6
STATE	no	no	no	yes|no	yes|no	yes
ACT	yes	yes	yes	no	yes	yes
ACC	yes	yes	yes	yes	yes|no	no
ACH	no	yes	no	no	no	no

In order to illustrate our procedure, let us take the verb *invoquer* 'to invoke' in one of the sentences where it appears in the corpus:

7. Pour justifier cette décision, la direction invoque la déprime du marché automobile.
'To justify this decision, the management invokes the depression of the automobile market.'

- **T1**: This verb cannot appear in a progressive form: **La direction est en train d'invoquer la déprime du marché automobile.*
- **T2**: *La direction a invoqué la déprime du marché automobile* is an acceptable answer to the question *Que s'est-il passé hier?*
- **T3**: This verb cannot appear as a complement of *commencer*, *continuer*, etc.: **La direction a commencé/continué à invoquer la déprime du marché automobile.*
- **T4**: *invoquer* is not compatible with *en x temps*: **La direction a invoqué la déprime du marché automobile en deux heures.*
- **T5**: the sentence is not compatible with *pendant x temps* either: **La direction a invoqué la déprime du marché automobile en deux heures.* This sentence is only acceptable in an iterative reading.
- **T6**: *La direction invoquait la déprime du marché automobile* does not imply *La direction a invoqué la déprime du marché automobile.*

Thus, according to the battery of tests summarized in Table 4, *invoquer* in (7) should be assigned to the ACHIEVEMENT class.

Table 4 Test results for (7).

	T1	T2	T3	T4	T5	T6
invoquer	no	yes	no	no	no	no

It is important to mention that verbs were annotated according to their meaning in the sentences found in the FTB corpus. Verbal polysemy was addressed only if different meanings appeared in the corpus. It is known that phrasal context can influence the verbal aspect ([8, 26] *inter alia*). Upon applying the tests presented above, plural subjects and direct objects were transformed into their singular forms, so as to avoid the effect that plural arguments can turn ACC predicates (*écrire un article en dix jours* 'to write a paper in ten days') into ACT ones (*écrire des articles pendant dix jours* 'to write papers for ten days'). Likewise, we have used past perfective tenses (*Elle a travaillé (hier)* 'She has worked (yesterday)') in order to avoid a habitual reading which is usually obtained in imperfective senses (*Elle (travaillait/travaille) à la poste* 'She (worked/works) at the post office'). Since imperfective tenses favour a habitual reading, the dynamicity property [±dynamic] of the verb becomes inaccessible. For similar reasons, frequency adverbs triggering iterative or habitual readings (*souvent* 'often', *tous les jours* 'every day') were not taken into account either, since they interfere with verbal aspectual features.

We obtain an aspectual characterization limited to the meanings appearing in the corpus. It is not an annotation of the verbs as lemmas, neither verbs in sentences, but rather an annotation of verbal structures (verb + arguments) in a discursive context, which allowed us to identify verbal meaning and to avoid polysemy as much as possible.

4.2 Annotation consistency assessment: Inter-Rater Reliability

Based on the annotation process outlined above, we have been able to estimate the inter-rater reliability (IRR), by taking into account the annotations produced by two annotators on 1161 verbs. The annotators are both experts in aspectual semantics. Each verb in the list has been annotated independently by each annotator, even though a final adjudication step yielded the annotations visible in the current version of the lexicon. Comparing the annotations produced by both annotators was necessary, in order to arrive at a consistent decision in the final resource. For example, *atteler* 'to tie' was initially labelled "ACT" by annotator 1, while annotator 2 was not sure of his annotation. After the first annotation phase, both annotators agreed to tag the entry as "ACT". Therefore, for the purpose of assessing the inter-rater agreement, we consider the initial annotation, which counts as a disagreement case. Conversely, for *cerner* 'to surround', annotator 1 was not sure of her annotation, while annotator 2 initially labelled the entry as "ACH". After confronting their annotations, both annotators finally agreed on labelling this entry as "ACC". Again, this case counts as a disagreement between both annotators. As can be seen, the final decision does not reflect either annotator's initial decision, which underlines the fact that aspectual annotation is a complex task. Cases such as the one discussed here therefore strongly advocate in favor of a post-annotation adjudication phase.

The following IRR statistics were produced using R packages: {irr}[7] and {irrCAC}[8]. Assessing IRR is not a straightforward task, since many methods have been presented in the literature[9]. We choose to present "standard" IRR statistics, such as Cohen's Kappa [5], in this preliminary stage, alongside Gwet's "Agreement Coefficient" score AC1 [14, 28]. Since the present lexical resource is still under construction, these IRR scores are essentially a way of assessing the complexity of the aspectual annotation task presented here, and therefore the consistency of the annotation procedure. In the annotation task under consideration, each annotator had to categorize 1161 verbal entries into 4 major classes: ACC, ACH, ACT, STATE. In total, 3 hybrid classes were also considered, such as: ACC/ACH, ACH/ACT and STATE/ACH. For example *varier* 'to vary' was initially labelled "ACH/ACT" by annotator 1 (final decision: "ACT"). Finally, a "not sure" tag was also used. As a consequence, the initial list of verbal entries has been associated with 8 different tags, including "not sure".

As can be seen in Table 5, both annotators agree on 82.6% of the cases, with an estimated 9.7% of chance agreement. The reported Kappa score (0.744) indicates a moderate inter-rater agreement[10], which is not uncommon for complex tasks. In our case, this score can be largely attributed to the fact that 4 major classes and 3 hybrid ones were considered. Gwet's AC1 score (0.806) is slightly higher than Cohen's Kappa, which can be attributed to

[7] Version 0.84.1, see [11] for more details on the underlying implementation, and [21] for a presentation of the R platform.
[8] Version 1.0, see [14] for a comprehensive presentation of the chance-corrected agreement coefficients implemented in this package.
[9] See [2] for a survey of IRR methods in NLP.
[10] Assessing the relevance of Kappa scores is known to depend heavily on the domain of application. We see these scores as an estimation of the task's complexity as well as the overall quality of the proposed annotations.

Table 5 IRR assessment of TreLex++ aspectual annotations.

Method	Score
irr (2 raters)	
unweighted Cohen's Kappa	0.744
irrCAC (confidence level = 0.95)	
percent agreement pa	0.826
percent chance agreement pe	0.097
AC1	0.806

the fact that Gwet's AC1 is a chance-corrected agreement coefficient that is known to yield higher agreement coefficients than Cohen's (and other authors'), in certain configurations. Regardless of the method, these figures indicate a "moderate" to "good" inter-annotator agreement.

At this point, it is worth emphasizing once more that, once the preliminary annotation was completed, a final adjudication phase took place, which yielded the final aspectual annotations visible in the current version of Treelex++. Since these final annotations are those end users will see, it is necessary to assess IRR scores between each annotator and the final annotations. In this case, Kappa scores in the 0.85 range, and AC1 scores in the 0.9 range can be reported. Final users of the TreeLex++ lexical resource should therefore consider that the proposed aspectual annotations are consistent, and that the annotation procedure based on syntactico-semantic tests achieves good results for the classes considered. As encouraging as they might seem, these figures should not obscure the fact that there is still considerable room for improvement, in terms of both scale and detail. For future versions, we are contemplating Games With A Purpose (GWAP) such as JeuxdeMots [17] as a source of user input. We are confident JeuxdeMots players will consider favorably new games, such as aspect-oriented tasks, provided we are able to propose 'gamified' versions of the present annotation procedure.

5 Data in TreeLex++

The resulting resource, TreeLex++, contains 1161 verbs enriched with syntactic (frame) and semantic (lexical aspect) properties. It is available in a text format as a CSV file (comma separated value). Each verb is accompanied by its frame, the lexical aspect, the number of examples found in the FTB and their full list[11]. To simplify the search of the inflected form in the example text, the corresponding verb is indicated between and tags, as presented in (8):

8. Quant à moi , je trouve qu' on se fiche du monde en n' expliquant pas les choses en langage courant .
'As for me, I think that they don't give a toss about the people by giving no explanation in the common language.'

To make linguistic generalizations easier, information encoded in syntactic frames has been translated into several representations:

[11] Individual examples are separated by a vertical bar '|'.

- number of syntactic arguments[12]
- whether a verb is reflexive or not
- a general frame (a list of syntactic functions and obligatory clitics)
- a simplified frame (a list of syntactic functions alone)
- the full frame including syntactic realizations (types of phrases)

The corresponding syntactic information for *déplorer* in (6) and the reflexive verb *se ficher* 'to not give a toss' presented in TreeLex++ format is given in Table 6.

Table 6 Syntactic information in TreeLex++.

Verb	Number of Arguments	Reflexive?	General frame	Simplified frame	Full frame
déplorer	2 2	no	SUJ.OBJ	SUJ.OBJ	SUJ:NP.OBJ: NP/Ssub
se ficher	2	yes	SUJ.DE-OBJ.refl	SUJ.DE-OBJ	SUJ:NP.DE-OBJ .refl:CL

A brief summary of syntactic realizations[13] of TreeLex++ verbs is given in Table 7 below. The number of arguments in TreeLex++ does not exceed three and the vast majority of verbs (74.24%) have two arguments. However, as indicated in Table 6, this does not necessarily correspond to a transitive structure (SUJ.OBJ) as the second argument may have a different function than a direct object (see Table 1).

Table 7 The distribution of verbs with respect to the number of arguments.

Number of Arguments	Total	Percent
1	183	15.76%
2	862	74.24 %
3	116	9.99%

The distribution of verbal aspectual classes found in TreeLex++ is given in Table 8.

Table 8 Aspect distribution in TreeLex++.

Aspectual class	Total	Percent
ACH	576	49.61%
ACC	260	22.39%
ACT	219	18.86%
STATE	103	8.87%
polysemous verbs	3	0.27%

The majority of verbs in TreeLex++ are telic (ACH or ACC). If we look at dynamicity, only a small proportion of verbs (8.87%) are true statives, the bulk of the entries are dynamic (ACH, ACC or ACT). However, the distribution of durative (STATE, ACT, ACC) and non-durative (ACH) verbs is almost equal.

[12] Clitic arguments are not considered here.
[13] The number of syntactic arguments.

The resource is neither syntactically nor semantically balanced, which is probably due to the content of the FTB corpus (newspaper texts).

As shown in Table 8, most verbs are assigned a single aspect. Hence, it seems that our approximate disambiguation technique is quite efficient. 3 verbs, however, exhibit a double aspect: *excéder*, *observer*, and *traverser*. Indeed, judging from their context, these verbs are truly polysemous in the FTB: *excéder* is ambiguous between 'to exceed' and 'to infuriate', *observer* is used as either 'to observe' or 'to respect/keep' and *traverser* corresponds to 'to cross' or 'to experience'. Therefore, even when syntactic properties are restricted to a single frame, certain semantic ambiguities could remain.

6 Conclusions and perspectives

TreeLex++ is a lexical resource which associates both syntactic and semantic properties, for over a thousand verbs, illustrated with attested examples taken from the FTB. Such a database offers a valuable resource for fundamental linguistics research, NLP and DH applications. From a fundamental research perspective, TreeLex++ allows to identify correlations, if any, between syntactic frames and aspect values. In other words, it allows researchers to work at the syntax/semantics interface. For instance, intuitively, the accomplishment verbs (ACC) should be associated with transitive verbs (2-argument predicates). TreeLex++ provides an opportunity to verify this hypothesis empirically: not only can it be confirmed or refuted but we can also estimate the degree of association between syntactic structures and aspectual classes. The first findings presented in [4] show how TreeLex++ can be put to use in this perspective. As for NLP applications, a number of practical uses of aspectual information is cited in [9]: the assessment of event factuality, text summarization, machine translation or automatic detection of temporal relations. We anticipate performance gains for those task, by integrating TreeLex++ as a symbolic resource, within a Machine Learning processing chain.

In its current version, TreeLex++ contains only single-frame verbs, which roughly covers a half of the entries in TreeLex. In order to include the remaining half in TreeLex++, we have to employ a true semantic disambiguation technique first. As mentioned in Section 5, a verb with a unique syntactic combination may still be polysemous and polyaspectual. In case of several frames, this potential ambiguity is multiplied and human disambiguation effort, already complex and time-consuming, increases considerably. A possible solution could be a lexical look-up of verb-frame couples in LVF [10] in order to identify different verb senses. However, pairing the senses with the corresponding FTB examples would require an ad-hoc approach. As mentioned above, another available option is to leverage user input, by resorting to crowd-sourcing, or "Game With A Purpose" platforms. We have taken steps towards this end by contacting Jeux de Mots's developer, Mathieu Lafourcade, in the perspective of integrating the aspectual information from TreeLex++ to the existing Jeux de Mots lexical network. This will allow for the development of new types of lexical games. We also hope Lafourcade's lexical propagation and integrity checking mechanisms will allow us to capture more general syntax/semantics properties than those which can be currently found in the FTB.

An evaluation methodology for our resource is also in order, beyond Inter-Rater Reliability scores, to determine the accuracy, as well as the coverage of our aspectual assignment process. For instance, we could compare our results with aspect values attributed to verbs in the Nomage project [3]. However, Nomage methodology (for verbs) differs from ours as aspect assignment is based on elicited examples rather than on verb uses in a corpus. Another

comparison could be made with the syntactico-semantic resource described in [9] which served for training of an automatic classifier of verbal aspect. Unfortunately, this data does not seem to be publicly available. Moreover, both resources use different aspectual values from ours thus the corresponding tagsets have to be converted first in order to provide the equivalent information. Again, we turn towards the Jeux de Mots platform, in the hope of gaining insights from users's inputs on lexical aspect assignment tasks[14], as well as from the network's built-in sanity checking mechanisms.

The current version of TreeLex++ is freely available on-line: `http://redac.univ-tlse2.fr/lexiques/treelexPlusPlus.html`. It can be either downloaded as a text (CSV) file or browsed directly via an intuitive on-line interface: `http://redac.univ-tlse2.fr/lexiques/treelexPlusPlus/interface/TreelexPlusPlusBrowser.html`.

References

1 Anne Abeillé, Lionel Clément, and François Toussenel. Building a treebank for French. In *Treebanks*, pages 165–187. Springer, 2003.

2 Ron Artstein and Massimo Poesio. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596, 2008.

3 Antonio Balvet, Lucie Barque, Marie Hélène Condette, Pauline Haas, Richard Huyghe, Rafael Marin, and Aurélie Merlo. La ressource Nomage. Confronter les attentes théoriques aux observations du comportement linguistique des nominalisations en corpus. *Traitement Automatique des Langues*, 52(3):129–152, 2011.

4 Antonio Balvet, Pauline Haas, Anna Kupść, and Rafael Marin. Looking for Syntax/Aspect Mappings: a Case Study on the French Treebank. In *Grammar and Corpora*, 2018.

5 Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

6 Anne Daladier. Le rôle des verbes supports dans un système de conjugaison nominale et l'existence d'une voix nominale en français. *Langages*, pages 35–53, 1996.

7 Marianne Djemaa, Marie Candito, Philippe Muller, and Laure Vieu. Corpus annotation within the French Framenet: a domain-by-domain methodology. In *Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, 2016.

8 David R Dowty. *Word Meaning and Montague Grammar: The Semantics of Verbs and Times in Generative Semantics and in Montague's Ptq*, volume 7. Springer Science & Business Media, 1979.

9 Ingrid Falk and Fabienne Martin. Automatic identification of aspectual classes across verbal readings. In *Sem 2016 The Fifth Joint Conference on Lexical and Computational Semantics*, 2016.

10 Jacques François, Denis Le Pesant, and Danielle Leeman. Présentation de la classification des verbes français de Jean Dubois et Françoise Dubois-Charlier. *Langue française*, 1(153):3–19, 2007. `doi:10.3917/lf.153.0003`.

11 Matthias Gamer, Jim Lemon, and Ian Fellows Puspendra Singh. *irr: Various Coefficients of Interrater Reliability and Agreement*, 2019. R package version 0.84.1. URL: `https://CRAN.R-project.org/package=irr`.

12 Howard B Garey. Verbal aspect in French. *Language*, 33(2):91–110, 1957.

13 Maurice Gross. *Méthodes en syntaxe*. Hermann, Paris, 1975.

14 Kilem L Gwet. *Handbook of inter-rater reliability: The definitive guide to measuring the extent of agreement among raters*. Advanced Analytics, LLC, 2014.

15 Anthony Kenny. *Action, emotion and will*. Routledge, 2003.

[14] In our view, the set of tests presented in Section 3 could very well be adapted to new games, focusing on aspectual properties.

16 Anna Kupść and Anne Abeillé. Growing Treelex. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 28–39. Springer, 2008.

17 Mathieu Lafourcade. Making people play for Lexical Acquisition with the JeuxDeMots prototype. In *SNLP'07: 7th International Symposium on Natural Language Processing*, page 7, Pattaya, Chonburi, Thailand, 2007. URL: `https://hal-lirmm.ccsd.cnrs.fr/lirmm-00200883`.

18 Mathieu Lafourcade and Alain Joubert. Détermination des sens d'usage dans un réseau lexical construit grâce à un jeu en ligne. In *TALN'08: Traitement Automatique des Langues Naturelles*, pages 189–199, 2008.

19 Béatrice Lamiroy. The complementation of aspectual verbs in French. *Language*, pages 278–298, 1987.

20 Joakim Nivre, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, and Deniz Yuret. The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 915–932, 2007.

21 R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013. URL: `http://www.R-project.org/`.

22 Susan Rothstein. *Structuring events: A study in the semantics of lexical aspect*, volume 5. John Wiley & Sons, 2008.

23 Benoît Sagot. The Lefff, a freely available and large-coverage morphological and syntactic lexicon for French. In *7th international conference on Language Resources and Evaluation (LREC 2010)*, 2010.

24 Benoît Sagot, Lionel Clément, Eric Villemonte de La Clergerie, and Pierre Boullier. The Lefff 2 syntactic lexicon for French: architecture, acquisition, use. In *LREC 06*, pages 1–4, 2006.

25 Zeno Vendler. Linguistics in philosophy. *Cornell University Press*, 1967.

26 Henk Verkuyl. A theory of aspectuality (cambridge studies in linguistics 64), 1993.

27 Marc Wilmet. Aspect grammatical, aspect sémantique, aspect lexical: un problème de limites. *J. David; R. Martin,(éds), La notion d'aspect, Metz: Centre d'Analyse Syntaxique de l'Université de Metz*, pages 51–68, 1980.

28 Nahathai Wongpakaran, Tinakon Wongpakaran, Danny Wedding, and Kilem L Gwet. A comparison of Cohen's Kappa and Gwet's AC1 when calculating inter-rater reliability coefficients: a study conducted with personality disorder samples. *BMC medical research methodology*, 13(1):1–7, 2013.

Annotation of Fine-Grained Geographical Entities in German Texts

Julián Moreno-Schneider
DFKI GmbH, Berlin, Germany

Melina Plakidis
DFKI GmbH, Berlin, Germany

Georg Rehm
DFKI GmbH, Berlin, Germany

Abstract

We work on the creation of a corpus, crawled from the internet, on the Berlin district of Moabit, primarily meant for training NER systems in German and English. Typical NER corpora and corresponding systems distinguish persons, organisations and locations, but do not distinguish different types of location entities. For our tourism-inspired use case, we need fine-grained annotations for toponyms. In this paper, we outline the fine-grained classification of geographical entities, the resulting annotations and we present preliminary results on automatically tagging toponyms in a small, bootstrapped gold corpus.

2012 ACM Subject Classification Information systems → Entity resolution; Information systems → Information extraction

Keywords and phrases Named Entity Recognition, Geographical Entities, Annotation

Digital Object Identifier 10.4230/OASIcs.LDK.2021.11

Supplementary Material *Collection (Collection of documents about Moabit district annotated with Geographical Entities)*: `https://gitlab.com/jmschnei/Moabit-Collection`

Funding The research presented in this paper is funded by the German Federal Ministry of Education and Research (BMBF) through the project QURATOR (`http://qurator.ai`) (Unternehmen Region, Wachstumskern, grant no. 03WKDA1A).

1 Introduction

The amount of information available in digital form is continuously growing, and a significant portion of it is accessed through mobile devices [2]. The fact that such devices are mobile in the first place, and usually also equipped with geolocation functionality, enables providing localised information to their users. A use case that can benefit from customised information, i. e., content tuned to the particular location, is tourism, e. g., an interactive travel guide, bringing points of interest in the vicinity to the user's attention [14, 15].

While exploiting a user's geographical location is relatively straightforward (privacy issues aside), combining this with information in textual form is less trivial. Typical corpora annotated for named entities, of which location is usually one of a small number of classes, do not distinguish between more detailed location-type entities, and, consequently, NER taggers do not make this distinction. We argue that for our use case of a travel guide, a more detailed distinction for toponyms, differentiating between, for example, (train/bus) stations, parks, streets and squares, is needed, allowing for more relevant, tailored recommendations. We explore this by defining a semantic classification of fine-grained geographical entities and by annotating a collection of documents accordingly. Our envisioned use case is to semi-automatically create a route or a guided tour, along which the user can explore the Berlin district of Moabit. This use case is to be understood in the context of the project

QURATOR[1], dealing with *digital curation technologies* [13]; promising results for a similar approach, but for a different domain, have been reported in [8, 9]. In QURATOR, we process large and multi-media document collections and analyse, re-arrange, summarise and visualise information contained in the collections, to generate stories – including the guided tours in our tourism use case – through *semantic storytelling*.

The rest of this paper is structured as follows. Section 2 describes related work focusing on the identification of geographical entities. Section 3 further motivates our work and explains the annotation guidelines. Section 4 describes the data we work with and provides results from first annotation efforts. Section 5 sketches a usage scenario for the annotated collection and reports on preliminary classification results. Finally, Section 6 sums up the main findings and provides pointers to future work.

2 Related Work

Specifically focusing on historical corpora, Won et al. [19] investigate the performance of five different NER systems for toponyms and spatial information. They find that using an ensemble method based on voting performs best. Similar to our envisioned approach, they experiment with combining NER modules with gazetteers. The use case of Alex et al. [1] is similar to ours. They use the Edinburgh Geoparser to tag and resolve fine-grained geographical locations in and around Edinburgh (both historical and contemporary). Also using the Edinburgh Geoparser, Gritta et al. [5] use two corpora (WikToR and LGL [4]) to evaluate five geo-parsers: CLAVIN[2], Yahoo!PlaceSpotter[3], The Edinburgh Parser [6], Topocluster [3] and GeoTxt [7] and conclude that the Edinburgh parser works best for them.

We adopt the idea of combining gazetteers (and also simple pattern-matching based components) with NER modules from earlier approaches, but the key feature that sets our use case apart from the ones mentioned above, is the fact that we include German as a language to detect more fine-grained geographical entities.

With regard to NER in general, we use and evaluate SpaCy[4], Stanford CRF NER[5] and an approach based on BERT[6], to recognise location-type entities. Out of the box and without specific re-training on a corpus annotated for more fine-grained types, they only output (among other types) locations, without any sub-classes.

3 Annotation of Fine-grained Geographical Entities

Many datasets for NER distinguish between persons, organisations and locations, and reserve a miscellaneous/other category for the remaining entities not captured by the former three categories [17, 18, 11]. In order to identify more specific (fine-grained) geographic entities, we need, first, a classification of the types of entities that we want to recognize, and, second, a collection of documents in which the entities are annotated based on that classification, that can be used as training data for new NLP models.

[1] https://qurator.ai
[2] https://clavin.bericotechnologies.com
[3] http://boss.yahoo.com, retrieved 2016-07-31.
[4] https://spacy.io
[5] https://nlp.stanford.edu/software/CRF-NER.shtml
[6] https://github.com/kamalkraj/BERT-NER

3.1 Semantic Classification

To annotate a corpus that further sub-classifies locations, we follow the NoSta-D guidelines [16]. We keep the four main categories of the NoSta-D-TagSet (`PER`, `LOC`, `ORG`, `OTH`; for persons, organisations, locations and other entities, respectively) and use the subcategories of the `LOC` category listed in the NoSta-D-TagSet as the foundation of our own fine-grained classification. As a result, a total of 14 fine-grained classes were developed for the `LOC` class (see Table 1). The other core NEs are kept (`PER`, `ORG`, `OTH`).

Furthermore, we follow the NoSta-D guidelines in that we do not annotate *dates*, *religions*, *names of animals*, *dynasties*, *cardinal directions*, *technical terms*, *salutations* or *political tendencies*. Additionally, as defined in the NoSta-D guidelines, we annotate categories such as *languages*, *websites*, *book/movie titles*, *wars* or *currencies* as belonging to the class `OTH`.

Table 1 Classification of subcategories for named entities, specifically for LOCATION entities.

Category	Label	Description	Examples
City	CTY	Capital cities, major and minor cities and smaller towns.	Berlin, Leipzig
Country	CNY	Based on the United Nations Member States [7].	Deutschland
State	STA	The 16 states (Bundesländer) of Germany.	Bayern, Brandenburg
Address	ADD	Toponym consisting of at least a street name and a number.	Unter den Linden 6, 10117 Berlin
Continent	CON	The seven continents.	Afrika, Australien
Building	BUILD	All buildings that are not considered as particular points-of-interest.	Anne-Frank-Grundschule, Johann von Neumann-Haus
Sight	SIGHT	Particular points-of-interest (i. e., popular sight-seeing locations)[8].	Brandenburger Tor, Siegessäule
Waters	WTR	Bodies of water such as lakes, rivers and channels.	Spree, Müggelsee
Address-Sub	ADD-SUB	Part of an address, not matching the address description.	Friedrichstraße, Potsdamer Platz
District	DST	Administratively recognised district or area of a city.	Mitte, Neukölln
Station	STN	Any train, bus or subway station.	U Turmstraße, S Friedrichstraße
Park	PARK	Parks and recreational areas.	Tiergarten, Britzer Garten
Shop	SHOP	Restaurants, cafés, bars and shops.	Sapori di Casa, Barcomi's Deli, George R
LOC-oth	LOC-OTH	Places that do not match any of the above.	Moltkebrücke
Org.	ORG	Companies, agencies, institutions, etc.	Apple, Samsung, Google
Person	PER	People, including fictional.	Angela Merkel
Other	OTH	All derived named entities (see NoSTa-D NEderiv), websites, book or movie titles, currencies, eras, languages, wars, etc.	www.google.de, Deutsch, Erster Weltkrieg

We deviate from the NoSta-D guidelines in that we do not use the *part* attribute for entities part of longer, complex tokens and we disregard the category `VLOC` (virtual location). We annotate derived entities (NEderiv in NoSta-D) as `OTH`, and finally we tag *restaurants*, *bars*, *cafes* and *hotels* as belonging to (sub-types of) locations, rather than organisations, because this better suits our tourism-inspired use case.

3.2 Integration of Linked Open Data

Annotated collections of documents are additionally improved through the inclusion of information from Linked Open Data (LOD) sources, to make them suitable for use with Linked Data approaches. For this we use a semi-automatic approach composed of two steps: first, we request two sources of information from which we automatically obtain URLs (looking up the entity on its (language-specific) DBpedia page using DBpedia spotlight [10]) and latitude and longitude (we use a SPARQL query against the Geonames[9] ontology) of the entities. Second, we extend this automatic process by validating the information obtained (because the correct information is not retrieved in all cases) and we complete manually the information in those cases in which it is missing (either URL or latitude/longitude).

4 Data and Annotation

Since our envisioned use case is in tourism, namely the generation of travel guides or guided tours, we decided to annotate a particular collection of documents, instead of taking a benchmark NER corpus and annotating our fine-grained location classes. The document set we used was collected through focused web crawling. We used Spidey[10] in combination with a list of manually generated seed terms, such as *Moabit*, *Kleiner Tiergarten* (a particular park in Moabit), *Kulturfabrik Moabit* (an event location) and *Kurt Tucholsky* (a German-Jewish author born in Moabit). In total, the complete list contains 28 items – places, buildings or persons – related to Moabit. This returned a list of URLs, which we crawled and boilerplated to extract the content and metadata using Newspaper3k.[11]

The resulting collection consists of 380 documents in German and 92 documents in English. We first tagged these documents with SpaCy and Stanford CRF-NER and proceeded with the Stanford CRF-NER output. After a manual revision, the documents contained 2682 (for German) and 777 (for English) `LOC` entities which we use as gold annotations. As a next step, we analysed the manually corrected `LOC` entities again and annotated them for the sub-classes listed in Section 3.1. The results are included in Table 2.

Examples 1 to 3 show three sentences extracted from the collection where several fine-grained geographical entities are annotated.

▶ **Example 1.** The site is openly accessible and you can stroll along the river Spree **WTR** .

▶ **Example 2.** To get to the AEG turbine factory **BUILD** from Hauptbahnhof **STN** , take the TXL **OTH** bus going to Tegel airport **LOC-OTH** and get off at Beusselstraße **STN** .

[7] `https://www.un.org/en/member-states/`
[8] Sights are a subcategory of buildings that are considered to be famous by the general population.
[9] `http://www.geonames.org`
[10] `https://github.com/vikrambajaj22/Spidey-Focused-Web-Crawler`
[11] `https://github.com/codelucas/newspaper`

Table 2 Absolute and relative frequency of each sub-class.

	German		English	
LOC sub-class	#	%	#	%
City	778	17.68	246	32
Country	678	15.41	14	2
State	90	2.04	2	<1
Address	295	6.70	13	2
Continent	32	0.73	0	0
Building	40	0.91	57	7
Sight	182	4.14	19	2
Waters	88	2.00	10	1
Address-Sub	251	5.7	37	5
District	441	10.02	32	4
Station	119	2.70	209	27
Park	111	2.52	2	<1
Shop	162	3.68	22	3
LOC-oth	1134	25.77	114	15
Total	2682	100	777	100

▶ **Example 3.** [...] of the author Hans Magnus Enzensberger **PER**, in Fregestraße 19 **ADD**, as well as in the studio apartment of the author Uwe Johnson **PER**, who was staying in the United States **CNY**, at Niedstraße 14 **ADD** in the Berlin **CTY** district of Friedenau **DST**.

As can be seen in Examples 4 and 5 the annotations still contain ambiguities. Example 4 has "Berlin" annotated as a city, although it could also be considered as incomplete, because the annotation could also include "North-West". If there are nested entities, we only annotate the longest entity which contains the nested ones.

▶ **Example 4.** Designed for Allgemeine Elektricitäts-Gesellschaft **ORG** in 1908 and constructed in 1910, it is located in North-West Berlin **CTY**, in the district of Moabit **DST**, around 3 Kilometers far from Reichstag **SIGHT**.

▶ **Example 5.** Since 1987, a memorial on the Putlitzbrücke **LOC-OTH**, which connects the districts of Moabit **DST** and Wedding **DST**, has commemorated the 30 Berlin **CTY** Jews who were deported from the nearby Moabit **DST** freight depot.

The annotated collection of documents is stored in a repository,[12] which is private to avoid any licensing issues (access can be granted upon request for research purposes). At the time of writing this paper, we are including the Linked Open Data information, which will be made available in the repository as an extended version of the collection.

[12] `https://gitlab.com/jmschnei/Moabit-Collection`

5 Geographical Entity Analyser

In order to demonstrate the functionality of the annotated document collection, we trained various named entity recognition modules using the collection.The current number of samples per class is too low to train a model on, but through our manual evaluation of the automatically tagged documents in the annotated document collection, we do have gold data for general `LOC` entities. To establish which approach performs best on this dataset for the coarse `LOC` entities, we compare three NER systems: SpaCy and Stanford (Section 2), and an approach based on BERT[13]. The SpaCy models are trained on Ontonotes 5 and Common Crawl (English; `en_core_web_md`) and WikiNER and TIGER (German; `de_core_news_md`). The Stanford models are trained on the CoNLL 2003 data [18]. BERT-NER is trained on WikiNER [11]. The results are shown in Table 3.

Table 3 Results for `LOC` entity recognition on the annotated document collection.

		Precision	Recall	F1
SpaCy	German	54.56	80.05	65.05
	English	77.94	54.57	64.19
Stanford	German	91.51	58.74	**71.55**
	English	84.75	50.06	62.95
BERT-NER	German	55.56	81.09	65.97
	English	70.71	59.97	**64.90**

Given that the Stanford output is the basis for our manual annotation, we expect a bias toward this system, and indeed we see that for German, this system outperforms the other two by approx. 6 points in F-score. For English however, the BERT-NER system performs best, though the difference with the other two is much smaller. Furthermore, having the initial character in uppercase is generally a distinguishing feature of named entities (and consequently a strong feature for many NER systems), but this indicator is not as strong in German, since nouns are by default in upper case. Still, the Stanford system performs considerably better for German than for English. The other systems do not exhibit the same disparity, and in fact perform better on English than on German. We consider looking into this an important venue for future work. Once we have annotated more data for the particular sub-classes, based on these intermediate results, we plan to train fine-grained modules, and perform a new comparison.

6 Conclusions and Future Work

We develop a dataset annotated with fine-grained geographical named entities (toponyms) that can be used to train named entity recognition modules that identify location-type entities in text documents. Regular NER modules typically distinguish four classes of entities (persons, organisations, locations, other). In our guided tour use case, knowledge about more detailed sub-classes allows for more relevant content and recommendations. In this first stage, we automatically tag a corpus crawled specifically for our use case and manually correct the output to obtain gold annotations for fine-grained entity types (i. e., bootstrap a small, gold

[13] `https://github.com/kamalkraj/BERT-NER`

corpus). Currently the dataset encompasses 2,682 (for German) and 777 (for English) `LOC` type entities. We report on performance for three NER systems on this dataset, although only using the coarse `LOC` type due to the size of the corpus.

In terms of future work, we plan to increase the volume of annotated data, both by annotating the remaining section of our crawled corpus and by double-annotating at least part of it to obtain inter-annotator figures. Once the corpus is in a more definitive state, we will examine how to make it available through the European Language Grid [12]. We will evaluate our model on the more detailed sub-classes using this final version of the corpus.

References

1 Beatrice Alex, Claire Grover, Richard Tobin, and Jon Oberlander. Geoparsing historical and contemporary literary text set in the City of Edinburgh. *Language Resources and Evaluation*, 53(4):651–675, 2019.

2 Irma Arts, Anke Fischer, Dominic Duckett, and René van der Wal. Information technology and the optimisation of experience – The role of mobile devices and social media in human-nature interactions. *Geoforum*, 122:55–62, 2021. doi:10.1016/j.geoforum.2021.03.009.

3 Grant DeLozier, Jason Baldridge, and Loretta London. Gazetteer-Independent Toponym Resolution Using Geographic Word Profiles. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 2382–2388. AAAI Press, 2015.

4 Milan Gritta, Mohammad Taher Pilehvar, and Nigel Collier. A Pragmatic Guide to Geoparsing Evaluation. *CoRR*, abs/1810.12368, 2018. arXiv:1810.12368.

5 Milan Gritta, Mohammad Taher Pilehvar, Nut Limsopatham, and Nigel Collier. What's missing in geographical parsing? *Language Resources and Evaluation*, 52(2):603–623, 2018.

6 Claire Grover, Richard Tobin, Kate Byrne, Matthew Woollard, James Reid, Stuart Dunn, and Julian Ball. Use of the Edinburgh geoparser for georeferencing digitized historical collections. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, 368:3875–89, August 2010. doi:10.1098/rsta.2010.0149.

7 Morteza Karimzadeh, Wenyi Huang, Siddhartha Banerjee, Jan Oliver Wallgrün, Frank Hardisty, Scott Pezanowski, Prasenjit Mitra, and Alan M. MacEachren. GeoTxt: A Web API to Leverage Place References in Text. In *Proceedings of the 7th Workshop on Geographic Information Retrieval*, GIR '13, page 72–73, New York, NY, USA, 2013. Association for Computing Machinery. doi:10.1145/2533888.2533942.

8 Elena Leitner, Georg Rehm, and Julián Moreno-Schneider. Fine-grained Named Entity Recognition in Legal Documents. In Maribel Acosta, Philippe Cudré-Mauroux, Maria Maleshkova, Tassilo Pellegrini, Harald Sack, and York Sure-Vetter, editors, *Semantic Systems. The Power of AI and Knowledge Graphs. Proceedings of the 15th International Conference (SEMANTiCS 2019)*, number 11702 in Lecture Notes in Computer Science, pages 272–287, Karlsruhe, Germany, 2019. Springer. 10/11 September 2019.

9 Elena Leitner, Georg Rehm, and Julián Moreno-Schneider. A Dataset of German Legal Documents for Named Entity Recognition. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Christopher Cieri, Khalid Choukri, Thierry Declerck, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020)*, Marseille, France, May 2020. European Language Resources Association (ELRA). Accepted for publication. Submitted version available as preprint.

10 Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. DBpedia spotlight: shedding light on the web of documents. In Chiara Ghidini, Axel-Cyrille Ngonga Ngomo, Stefanie N. Lindstaedt, and Tassilo Pellegrini, editors, *I-SEMANTICS*, ACM International Conference Proceeding Series, pages 1–8. ACM, 2011. URL: http://dblp.uni-trier.de/db/conf/i-semantics/i-semantics2011.html#MendesJGB11.

11 Joel Nothman, Nicky Ringland, Will Radford, Tara Murphy, and James R. Curran. Learning Multilingual Named Entity Recognition from Wikipedia. *Artif. Intell.*, 194:151–175, 2013. doi:10.1016/j.artint.2012.03.006.

12 Georg Rehm, Maria Berger, Ela Elsholz, Stefanie Hegele, Florian Kintzel, Katrin Marheinecke, Stelios Piperidis, Miltos Deligiannis, Dimitris Galanis, Katerina Gkirtzou, Penny Labropoulou, Kalina Bontcheva, David Jones, Ian Roberts, Jan Hajic, Jana Hamrlová, Lukáš Kačena, Khalid Choukri, Victoria Arranz, Andrejs Vasiļjevs, Orians Anvari, Andis Lagzdiņš, Jūlija Meļņika, Gerhard Backfried, Erinç Dikici, Miroslav Janosik, Katja Prinz, Christoph Prinz, Severin Stampler, Dorothea Thomas-Aniola, José Manuel Gómez Pérez, Andres Garcia Silva, Christian Berrío, Ulrich Germann, Steve Renals, and Ondrej Klejch. European Language Grid: An Overview. In Nicoletta Calzolari et al., editor, *Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020)*, pages 3359–3373, Marseille, France, 2020. European Language Resources Association (ELRA).

13 Georg Rehm, Peter Bourgonje, Stefanie Hegele, Florian Kintzel, Julián Moreno Schneider, Malte Ostendorff, Karolina Zaczynska, Armin Berger, Stefan Grill, Sören Räuchle, Jens Rauenbusch, Lisa Rutenburg, André Schmidt, Mikka Wild, Henry Hoffmann, Julian Fink, Sarah Schulz, Jurica Seva, Joachim Quantz, Joachim Böttger, Josefine Matthey, Rolf Fricke, Jan Thomsen, Adrian Paschke, Jamal Al Qundus, Thomas Hoppe, Naouel Karam, Frauke Weichhardt, Christian Fillies, Clemens Neudecker, Mike Gerber, Kai Labusch, Vahid Rezanezhad, Robin Schaefer, David Zellhöfer, Daniel Siewert, Patrick Bunk, Lydia Pintscher, Elena Aleynikova, and Franziska Heine. QURATOR: Innovative Technologies for Content and Data Curation. In Adrian Paschke, Clemens Neudecker, Georg Rehm, Jamal Al Qundus, and Lydia Pintscher, editors, *Proceedings of QURATOR 2020 – The conference for intelligent content solutions*, Berlin, Germany, 2020. CEUR Workshop Proceedings, Volume 2535. 20/21 January 2020.

14 Georg Rehm, Karolina Zaczynska, Peter Bourgonje, Malte Ostendorff, Julián Moreno-Schneider, Maria Berger, Jens Rauenbusch, André Schmidt, Mikka Wild, Joachim Böttger, Joachim Quantz, Jan Thomsen, and Rolf Fricke. Semantic Storytelling: From Experiments and Prototypes to a Technical Solution. In Tommaso Caselli, Eduard Hovy, Martha Palmer, and Piek Vossen, editors, *Computational Analysis of Storylines: Making Sense of Events*. Cambridge University Press, 2021. In print.

15 Georg Rehm, Karolina Zaczynska, Julián Moreno Schneider, Malte Ostendorff, Peter Bourgonje, Maria Berger, Jens Rauenbusch, André Schmidt, and Mikka Wild. Towards Discourse Parsing-inspired Semantic Storytelling. In Adrian Paschke, Clemens Neudecker, Georg Rehm, Jamal Al Qundus, and Lydia Pintscher, editors, *Proceedings of QURATOR 2020 – The conference for intelligent content solutions*, Berlin, Germany, 2020. CEUR Proc., Vol. 2535. 20/21 Jan. 2020.

16 M. Reznicek. Linguistische Annotation von Nichtstandardvarietäten: Guidelines und Best Practices: Guidelines NER. Technical report, Humboldt-Universität zu Berlin, September 2013.

17 Erik F. Tjong Kim Sang. Introduction to the CoNLL-2002 Shared Task: Language-Independent NER. In *Proceedings of CoNLL-2002*, pages 155–158. Taipei, Taiwan, 2002.

18 Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, page 142–147, USA, 2003. Association for Computational Linguistics. doi:10.3115/1119176.1119195.

19 Won, Miguel and Murrieta-Flores, Patricia and Martins, Bruno. Ensemble Named Entity Recognition (NER): Evaluating NER Tools in the Identification of Place Names in Historical Corpora. *Frontiers in Digital Humanities*, 5:2, 2018. doi:10.3389/fdigh.2018.00002.

Supporting the Annotation Experience Through CorEx and Word Mover's Distance

Stefania Pecòre
School of Electrical Engineering and Computer Science, University of Ottawa, Canada

Abstract

Online communities can be used to promote destructive behaviours, as in pro-Eating Disorder (ED) communities. Research needs annotated data to study these phenomena. Even though many platforms have already moderated this type of content, Twitter has not, and it can still be used for research purposes. In this paper, we unveiled emojis, words, and uncommon linguistic patterns within the ED Twitter community by using the Correlation Explanation (CorEx) algorithm on unstructured and non-annotated data to retrieve the topics. Then we annotated the dataset following these topics. We analysed then the use of CorEx and Word Mover's Distance to retrieve automatically similar new sentences and augment the annotated dataset.

2012 ACM Subject Classification Applied computing → Document management and text processing; Applied computing → Annotation

Keywords and phrases topic retrieval, annotation, eating disorders, natural language processing

Digital Object Identifier 10.4230/OASIcs.LDK.2021.12

Funding IT12441 MITACS and SafeToNet Canada

Acknowledgements We thank MITACS and SafeToNet Canada for their generous funding. In addition to this, we thank the University of Ottawa and the supervisor of the project, Professor Diana Inkpen, for their support.

1 Introduction

Online social platforms provide an easy way to share ideas, opinions, information, and personal messages. Research suggests that online communities are a support tool for recovery and promotion of self care and well-being [21]. At the same time, these platforms may be used to enhance and promote destructive behaviours as in pro-Eating Disorder communities (pro-ED groups). Eating disorders such as *anorexia nervosa, binge eating disorder, and bulimia nervosa* are recognized as mental disorders in standard medical manuals (ICD-10[1] and DSM-5[2]). The exact etiology of eating disorders remains unclear [35, 13] and they are a real concern due to the highest mortality rate of any mental illness, affecting various ethnic groups [28], males and females [15], any age range [16], with a highest peak during teen age [3]. During the last 10 years the research community has been analysing pro-ED groups using different platforms: Instagram [9, 8], Tumblr [14], Flickr [46], Reddit [32, 38], YouTube [39], Twitter [1, 45]. The analyses have been carried out according to different points of view: social media moderation [7, 9], relation between pro-ED users and ED content [1, 34], contrast between similar – "thinspiration" and "fitspiration" [41], online ED content analysis [5, 48, 4, 19, 40], pro-ED users' identity perception [2, 20], ED markers [33], multimodal classification [7], and early detection of anorexia signs [42]. ED and mental illnesses have also been the main focus of recent workshops such as CLEF E-risk and

[1] https://www.who.int/classifications/icd/icdonlineversions/en/
[2] https://www.psychiatry.org/psychiatrists/practice/dsm
[3] https://www.eatingdisorderhope.com/information/statistics-studies

CLPsych [27, 26, 25, 12]. According to Twitter rules and policies[4], it is not possible to promote or encourage self-harming behaviours (including eating disorders). However, Twitter has not banned or restricted the access to any specific pro-ED related hashtags and content, allowing us use this data for research purposes.

The main contributions of this paper are: (a) a new use of Correlation Explanation (CorEx) algorithm [44] to retrieve topics, emojis, and contextual foreign words in English tweets of native and non-native English Eating Disorder communities, (b) the use of the Word Mover's Distance (WMD) model [24] to annotate similar sentences and assist the annotators. We aim to create a tool to assist annotators in their annotation task, by providing a way to semi-automatically increment the number of annotated sentences, even in a complex context such as the ED communities. To the best of our knowledge, the amount of models annotating 10 years of tweets in emojis and non-common word patterns related to Eating Disorders (ED) with the use of CorEx to extract ED topics [48, 17] and Word Mover's Distance to assist annotators is limited at this time. We believe that this work could be of interest for the research community also in other domains, where topic extraction and annotation are involved.

We will describe our dataset (Section 2), then we will present the CorEx algorithm (Section 3) and the reasons behind the choice of this algorithm instead of others frequently used such as LDA. We will explore how we used CorEx to retrieve documents correlated with the topics (Section 4) and, consequently, why and how we decided to manually annotate our dataset for ED aspects (Section 5). Finally, we will describe our approach with Word Mover's Distance to assist annotators in data annotation tasks (Section 6), and we will identify the limitations of this work (Section 7), followed by conclusions and future work.

1.1 Ethical Considerations

This work uses public tweets from 2009 to 2019. No personally identifiable information (location, photos, names) was used in this study, nor was included in any of our algorithms. We did not interact with the subjects of this study, and since the data is public, we did not need institutional review board approval. The annotators were given anonymized data.

2 General pro Eating Disorder (pro-ED) Twitter dataset

2.1 Data Collection

In order to create our initial dataset, we collected tweets by using known ED tags [8, 14, 6] through a library[5] preserving Unicode emojis. From the seed tags, we retrieved both related posts and ED hashtags (a partial list is shown in table 1). We found low frequency hashtags that were not related to the ED and others that – without a context – seemed not directly related to ED (#casuloana, #whale, #borboletana) and their presence became more clear during the annotation phase. The datasets are in English and available under request – due to NDA reasons.

[4] https://help.twitter.com/en/rules-and-policies/glorifying-self-harm
[5] https://pypi.org/project/GetOldTweets3/

Table 1 Examples of hashtags found after retrieving the data.

Keywords Found	No. of # in the dataset	Keywords	No. of # in the dataset
#thinspo	78,244	#skinny	7,446
#proana	38,692	#ana	6,213
#thinspiration	10,471	#weightloss	5,322

2.2 Data preprocessing

The initial dataset was composed of 106,793 tweets dated from 2009 to 2019. During the normalization phase we transformed words in lowercase and removed most of the non-ED related tweets – e.g. tweets that had more than 80% of the content about link referrals. We also removed duplicates and punctuation (except the symbol # to preserve the hashtags). We applied the fastText Language Identification tool [23, 22] as a filter to avoid non-English sentences. We applied a basic anonymization filter by replacing tagged user, with the corresponding label USER, numbers with the label NUM, websites with the label URL, common cities with the label LOC. After the normalization and anonymization phases, our dataset had 87,957 entries.

From the analysis of our dataset (Table 2 & Table 3) we noticed a low lexical diversity: there are only 67,296 types[6] over 1,150,508 tokens in total. Moreover, the pro-ED community on Twitter seems to prefer writing on average short tweets: less than 11 words per tweet and 41–60 characters (Figure 1). We expected that after Twitter's characters doubling in 2017, the most recent tweets would have been longer. This was the case only for 1% of the tweets.

Table 2 Dataset lexical analysis.

No of tweets	No of tokens	No of types	Type/Token ratio
87,957	1,150,508	67,296	5,85%

Table 3 Average, Median and Standard Deviation of Words and Characters per tweet.

Distribution of WORDS per tweet		Distribution of CHARACTERS per tweet	
AVG	11.8	**AVG**	75.2
MDN	10	**MEDIAN**	65
STDEV	7.04	**STDEV**	42.3

2.3 Emojis

In order to have emojis present later in our tests, we decided to translate them from Unicode to their description. The description has been taken from the CLDR Short Name information repository[7]. For example:

- the *Unicode U+1F600* corresponds to the description *grinning face*,
- the *Unicode U+1F605* corresponds to the description *grinning face with sweat*.

We noticed a low frequency in conjunction with a low diversity in the use of the emojis: only 11.75% of the lines showed one or more emojis and only 4.75% of emoji types have been used.

[6] `https://plato.stanford.edu/entries/types-tokens`
[7] `https://Unicode.org/emoji/charts/full-emoji-list.html`

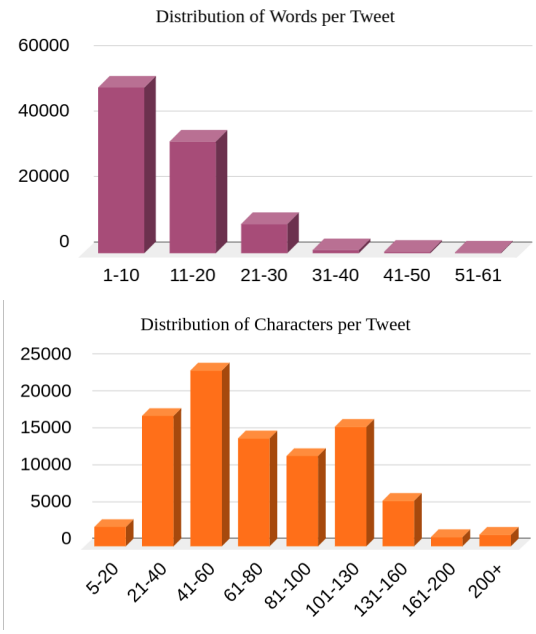


Figure 1 Distribution of words and characters per tweet.

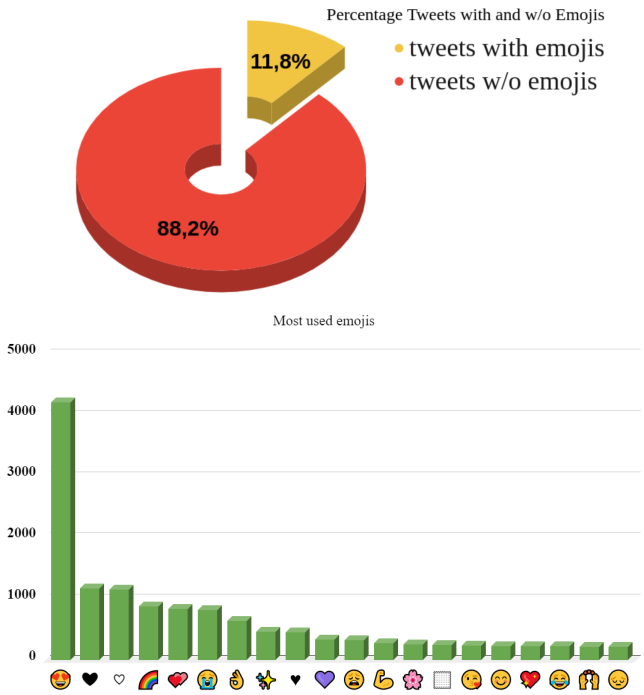


Figure 2 Distribution of emojis in the dataset.

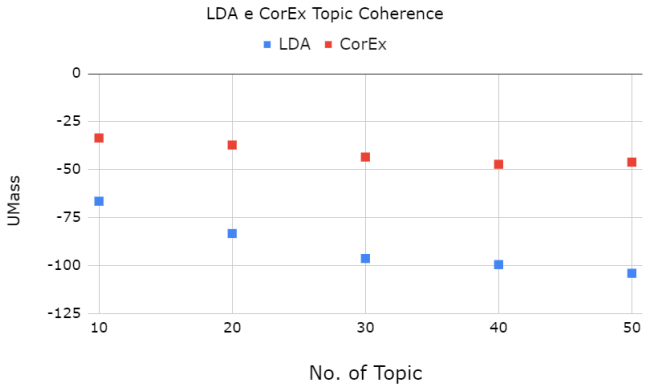


Figure 3 Comparison of CorEx to LDA with respect to topic coherence on ED Dataset.

We noticed that only 15% of the most used emojis seem to convey a negative sentiment. We remarked that positive polarity emojis don't pair always with an overall positive content: the first most used emoji, *face with heart shaped eyes*, has been used for the appreciation of emaciated bodies. Even though emojis will be present on our experiments and results, they are only a part of our work, analysis, and findings on this type of communication.

3 Finding Eating Disorder related topics through CorEx

3.1 Why did we choose CorEx?

Research has already explored the use of topic modeling to assist document annotation [43, 37, 47, 11, 48]. For our experiments we needed a model able to extract topics in unstructured and low-diversity data (see subsection: 2.2). According to the authors of CorEx [17], the model should work better than LDA models [3], since it maximizes the mutual information between words and topics without any assumption on how documents are generated [17]. Plus, according to the authors, CorEx is a discriminative model that works very well with minimal domain knowledge, which is the case when pre-annotated data is not present. In order to test whether CorEx works better than LDA also for our specific dataset, we tested it against LDA in detecting semantic topic quality as in [18]. As the authors wrote, CorEx does not explicitly attempt to learn a generative model and, traditional measures such as perplexity are not appropriate for model comparison against LDA. Furthermore, it is well-known that perplexity and held-out log-likelihood do not necessarily correlate with human evaluation of semantic topic quality [10, p. 6]. For this reason, we measured the semantic topic quality using Mimno et al.'s [31] UMass topic coherence score, which correlates with human judgments.

We used the same dataset for both models and we varied the number of topics. We ran the model 30 times for each time we changed the number of topics. We tested the models using 10, 20, 30, 40, 50 topics. For both models, we noticed the tendency to obtain worse topic coherence when the number of topics increased – see Figure 3. Each dot is the average of 30 runs. We suppose that the low lexical variety is due to the small number of topics discussed. For both models, the optimal number of topics seems to be 10. When we compared CorEx to LDA, we noticed that CorEx outperformed LDA in terms of topic coherence. For this reason, we decided to use CorEx to identify and describe the latent topics from our dataset. The final number of chosen topics is 10.

Once the number of topics has been chosen, two experts manually reviewed the words of the topics learned by CorEx to re-verify the coherence and the content. The categories, their description, and some examples are shown in Table 4. The results of the evaluation confirmed the ED topics and keywords that have been described in other ED related studies [8, 14, 6].

3.2 Results from CorEx application

During the manual review of the extracted words, we found both emojis and special words connected to the topics. We decided purposely not to remove them, because we believe that emojis, as well as special words, could indicate a presence of eating disorder content.

We carried out an in-depth analysis of special words that are relevant to pro-ED communities. It seems interesting to highlight that, although they seem not relevant, they are uncommon contextual words. Following Table 4, here are some interesting findings:

1. **foreign language keywords** – from row ED OTHER LANGUAGES: even though we removed non-English sentences, there are some sentences that are written in English, with hashtags or a partial content in another language. We were able then to capture hidden contents also in other languages. An example is the hashtag #waniliowemleko ["*vanilla milk*" in Polish] that has been found also in some pro-ED websites[8] citing a popular drink within the ED community, that seems to be used as a social drink with few calories. We noticed that these words are usually associated with other ED English relevant keywords, such as *#skinny* and *#diet*;
2. **acronyms** – from DIET row: we noticed that the model was able to capture words that may appear of difficult interpretation without prior knowledge and a context. For examples: *NF* for "*no food*", *OMAD* for "*One Meal A day*", *ABCDIET* for "*Ana Boot Camp diet*", *NT* for "*no thanks*" (usually linked to food offer as in "*dinner? NT*");
3. **celebrities** – in row SPORT: there are references to some YouTube celebrities ("*Lena Snow*", "*Chloe Ting*", "*Alexis Ren*") who stream weekly workouts;
4. **ED slang**: the model also unveiled other words that may not seem significant without prior knowledge, but they refer to encrypted community slang. We refer to words such as "*borboletana*" (from "*borboleta*", butterfly in Portuguese, a shared symbol of pro-ED and recovery communities, and "**AN**orexi**A**"), "*casuloana*" ("*casul of Ana*", from "casual" of ana that in Internet slang means newbie of ana),"*rexy*" ("ano**RE**xia + se**XY**") and *#skinnylegend* (which represents skinny photoshopped celebrities' body);
5. **relevant ED emojis**: we found that the emojis issued from the model are in line with the words associated and they:
   a. **may reinforce the meaning of the word** – from the row CONSEQUENCES: the emojis [mouth], [nauseated face], [face with open mouth vomiting] seem to be properly associated with words such as *"purge"*, *"binge"*, *"puke"* that appear in the dataset;
   b. **may publicly manifest user's gender**, such as [female sign] in GENERAL ED
   c. **may express sarcasm**, such as [face upside down] found in WEIGHT row. This emoji is present in sentences like "*yesterday I ate like a normal person cus I was with my family and woke up 1.2 lbs heavier [face upside down] fuck*" and "*I'm going on a binge. Can't wait to purge [face upside down] I'm such a fat ass.*")
   d. **may express more than words**, such as [dizzy symbol] in SPORT row.

[8] `https://www.wattpad.com/334686866-sad-skinny-girl-guide-eating-out-starbucks`

Table 4 Some examples of retrieved topics with their descriptions, related words, and emojis found (the ratio between content words and hashtags displayed here is not representative of the distribution of this dataset.)

Topic Category	Description	Words	Emojis
ED words in other languages	not English words related to ED	#abwtbs #bslyw #waniliowemleko #caspfb38 #samajl	NA
GENERAL ED	General References to pro-ED content	#proana #atypicalanorexia #slimthickspo #casuloana #edtwt #edtwitter #edproblems	[face_with_tears_of_joy] [female_sign] [butterfly] [person_shrugging] [face_with_pleading_eyes]
WEIGHT	Everything related to the person's weight and weight management	lose weight lbs gain pound #goal #weightcheck cw ugw sw #bodycheck	[face_upside_down]
BODY REPRESENTATION	how they judge themselves compared to another body not in a measurable way	#butterfly #borboletana #dysmorphia #fattie #fatpig #rexy	[butterfly] [broken_heart]
BODY DESCRIPTION	how they describe a body in a neutral manner	bone collar collarbone hip #hipbone thigh gap flat waist cm #fat	NA
SELF REPRESENTATION	How they judge themselves in a not measurable way	bitch cow #whale stupid whore ugly #ass cunt dumb #lazy #pathetic	[whale]
HARM	Everything related to self harm and risky consequences for the person	#depression #anxiety #selfharm #selfhate #cutting #deadinside laxative pills	[face_with_medical_mask] [knife
COMMUNITY	Interactions within the pro-ED group	#meanspocoach #sweetspo #bonespo #nicespo #skinnylegend #malespo rexy	[smiling_face_with_heart] [smiling_face_with_smiling_eyes] [white_heart]
DIET	Everything related to diet and calories	#stopeating NT NF #donteat #foodfears #OMAD fasting #abcdiet calorie intake	[raised_fist]
SPORT	Everything related to sport and activities to burn calories	workout lena snow chloe ting alexis ren gym routine #itsallfine	[thumbs_up_sign] [dizzy_symbol] [skull_and_crossbones] [flex_biceps]
CONSEQUENCES	person's action impacts on him/her life	stomach hurt pain growl grumble purge force parent haven	[mouth] [nauseated_face] [face_with_open_mouth_vomiting]

4 Automatic Retrieval of documents through CorEx

We then used an internal function of CorEx to retrieve the documents with the topics discovered before, because we believe that using CorEx to retrieve topics and also documents at the same time could make the annotation process faster and smoother. We retrieved the most probable documents per topic. According to the paper [18], CorEx estimates the logarithmic probability of a document belonging to a topic given that document's words. In order to evaluate this process, we retrieved the first 100 most probable documents for the topics *Weight, Body Representation, Self Representation, Harm, Consequences, Community, Sport, Diet, Body Description, and General ED*. We then created guidelines explaining the type of topic described by CorEx using some examples, and we asked two native English speakers to evaluate whether a sentence belonged to a certain topic or not. Except for the topic *General ED*, we noticed that the results were not satisfying (see Table 5). Since we were not able to retrieve the documents directly from the estimation of the probability of CorEx, we decided to use another algorithm to discover, given a seed of annotated documents this time, other documents similar to the annotated ones. The chosen algorithm is Word Mover's Distance (WMD) [24]. We chose this algorithm because it targets both semantic and syntactic information to calculate similarity between text documents. It is designed to overcome the synonym problem: since similar words should have similar vectors, WMD can calculate the distance even when there are no common words.

In order to have the same type of dataset to evaluate this method, we chose to annotate a part of the pro-ED dataset. Then, we ran the experiments against this dataset in a controlled environment.

Table 5 F1-score for 100 most-probable documents belonging to the topic.

	Weight	Body Representation	Self Representation	Harm	Consequences	Community	Sport	Diet	Body Description	General
F1	0.0178	0.1291	0.0159	0.0275	0.0315	0.0676	0.0334	0.0564	0.1326	0.6412

5 Annotation scheme and examples

The annotation scheme is composed of ten categories. We decided to remove the *ED words in other languages* category for simplification since the tweets were not completely written in English. Here are some examples for each category:

1. **Body Description (BD)**: "I **am** skinny"
2. **Body Representation (BR)**: "I **want** to be skinny **as** her"
3. **Community (COM)**: "I love my #anasisters"
4. **Consequences (CON)**: "Finally today i eat And a feel a little bit bad"
5. **Diet (D)**: "tomorrow I'm going to start fasting again"
6. **General ED (G)**: "Being thin and not eating are signs of true will power and success #proana"
7. **Harm (H)**: "I don't like laxatives but it's time"
8. **Self Representation (SR)**: "I am perfect", "I woke up and I was still UGLY thanks for nothing"
9. **Sport (S)**: "Insanity kicked my butt What a good workout"
10. **Weight (W)**: "Losing weight is good, gaining weight is bad"

We would like to make three specific categories explicit, as we did before the beginning of the annotation phase with the annotators, as they may be cause of confusion:

- **Body description** is applied to each sentence where there is a statement about a body (where the person describes the body). Examples are: *"she's so skinny"*, *"I am skinny"*, *"look at her collarbones!"*
- **Body representation** is applied to each sentence where the people refer to themselves by means of a comparison with other people. Examples are: *"I want to be skinny as her"*, *"I want her legs"*
- **Self representation** is applied to each sentence where people judge themselves using not measurable and imaginary words. Examples are: *"I'm ugly"*, *"I'm perfect"*, *"I would like to be graceful as a butterfly"*

5.1 Human Annotation process and guidelines

The annotation has been done via PigeonXT[9] by two English native speakers. The annotation was done at the sentence level to identify the topics. In total, 3,064 sentences (Table 6) have been annotated. Even if a tweet can be as short as the average of 11 words, we assumed that it was possible to find more than one category per tweet. However, we decided to take into consideration only one category at a time, considering the overall content complexity. At the

[9] `https://github.com/dennisbakhuis/pigeonXT`

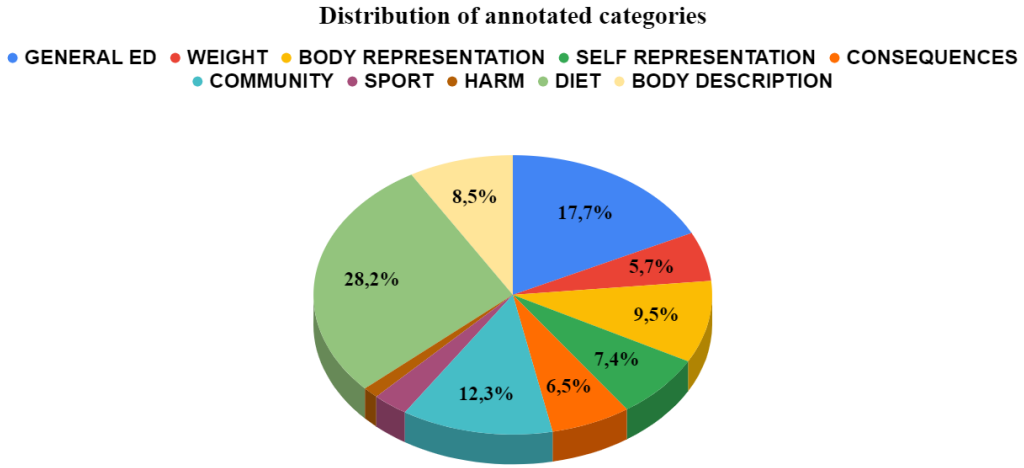


Figure 4 Annotations distribution.

end of the annotation task, we measured Cohen's Kappa, and it was 0.89 between the two annotators. We believe this is acceptable given the difficulty of the task. A set of rules and examples has been given to the annotators:

1. For each sentence they could choose up to two categories among those available;
2. When they found two categories, they could choose a main one and a secondary one, if necessary. Example: "*I feel so clean right now no food in a week #ProAna!*" has been annotated as primary DIET ("*no food in a week*") because the tweet is about not having food in a week and with the consequences of feeling clean, so the secondary will be SELF REPRESENTATION ("*I feel so clean*");
3. Sometimes Twitter users' use hashtags to index the content and have more chances to have their profile found. For this reason annotators distinguished hashtags between (a) unit of content and replaceable with the same word (example: "*I am happy to be in my #proana club!*"), and (b) – usually – a sequence at the end of the sentence used only to index the content and not relevant for the topic expressed (example: *"ugh! #proana #edtwt #ed #anabuddy #weightloss*").
4. Whenever it was not possible to label specific category and if the sentence was still related to ED they could use the GENERAL ED category.

A cross-reading has been done to improve the reliability of the dataset annotatation.

5.2 Annotation results and discussion

Table 6 Annotated sentences.

No. of sentences	**3,064**
No. of words	20,838
No. of types	5,528

By analyzing the annotation results (see Figure 4) we noticed three major categories: DIET, GENERAL ED, and COMMUNITY. We think that these results reflect the major characteristics of this community: their worries about what they eat (DIET), their need to have a group to whom to talk and share (COMMUNITY). Finally GENERAL ED regroups everything that may be shared online and not necessarily being confined in a specific category. This highlights also the wealth of arguments of this type of user.

We noticed that the most difficult sentences to annotate have been the ones that would be ambiguous without any further context, and the ones showing more than two categories at the same time, such as:

- "being *hungry asshat* until you get tired of it... *Die* FFS!": this sentence could be annotated as CONSEQUENCES (*being hungry*) SELF REPRESENTATION (*asshat*) HARM (*Die*);
- "*feeling fat*" versus "*be fat*": here annotators agreed to annotate the first as SELF REPRESENTATION and the second as BODY DESCRIPTION.

6 Word Mover's Distance to annotate similar sentences

In order to evaluate the use of Word Mover's Distance (WMD) for annotation, we used the same annotated dataset.

Word Mover's Distance is an adaption of Earth Mover's Distance (EMD) [36] which uses word embeddings to determine the similarity between two or more series of words (e.g., sentences). WMD uses the locations and words relative frequency weights of word embeddings to find the nearest neighbor for each word. Specifically, WMD uses the product of two numbers: the cosine distance between two words in the n-dimensional embedding space, and a weighting term that indicates how much one word in one document must travel to another word in the other document. In this way, it can minimize the cost of moving all words from a document to the positions of all the words in another document. The documents that share many semantically-similar words will have smaller distances than the documents with dissimilar words. In Figure 5, we show four sentences, originally from [24]:

1. D0: The President greets the press in Chicago
2. D1: Obama speaks to the media in Illinois
3. D2: The band gave a concert in Japan
4. D3: Obama speaks in Illinois

The relative cost of moving all the words in D2 to the locations of the words in D0 is greater than moving the words in documents D1 and D3. Formally:

$$WMD_{ij} = min_{T \geq 0} \sum_{i,j=1}^{2} T_{ij} c(i,j)$$

$$\sum_{j=1}^{2} T_{ij} = d_i, \forall i \in \{1, ..., n\}$$

$$\sum_{i=1}^{2} T_{ij} = d'_j, \forall j \in \{1, ..., n\}$$

with c(i,j) representing the euclidean distance $||\mathbf{x_i} - \mathbf{x_j}||_2$ between the two words in the embedding space. The travel cost between two words translates in the distance between texts. Let $\mathbf{d}$ and $\mathbf{d'}$ be the documents with each word i in $\mathbf{d}$ to be transformed into any words inside the document $\mathbf{d'}$. $\mathbf{T}$ is the sparse matrix where $\mathbf{d'}$ represents how much of i in $\mathbf{d}$ travels to word j in $\mathbf{d'}$. We expect that the moving from word i must equal to $\mathbf{d_i}$ in order to allow the transformation of $\mathbf{d}$ into $\mathbf{d'}$. The same is applicable for the word j that should match $\mathbf{d'_j}$.

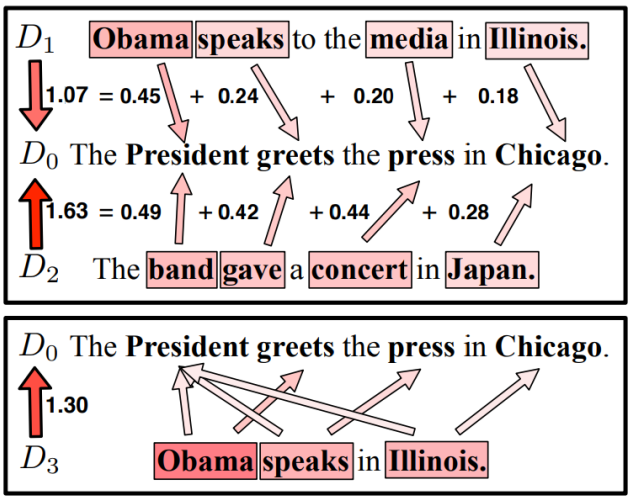


Figure 5 Illustration of Word Mover's Distance from Kusner et al. [24].

Our goal was to evaluate whether WMD could be used to improve the annotation process by verifying that the most similar sentences retrieved by WMD were of the right class. First of all, we trained Word2vec [30, 29] using the Gensim package [10] on the whole annotated dataset with vector size equal to 100. Then, we isolated 30% of the sentences for each class, and we run WMD against them. Finally, we retrieved the most similar sentences for each sentence of that 30% with a threshold of similarity of 0.98 and above, excluding the sentences used to compute the similarity. We evaluated this method by comparing the most similar sentences per class with their real class labels. Our results are shown on table 7. They show that WMD is a good model compared to CorEx to annotate new sentences when similarity is the searched parameter.

Table 7 F1 score for the sentences retrieved using Word Mover Distance.

	Weight	Body Representation	Self Representation	Harm	Consequences	Community	Sport	Diet	Body Description	General
F1-Score CorEx	0.0178	0.1291	0.0159	0.0275	0.0315	0.0676	0.0334	0.0564	0.1326	**0.6412**
F1-Score WMD	**0.7686**	**0.7909**	**0.7544**	**0.51**	**0.9721**	**0.8319**	**0.7404**	**0.7756**	**0.6114**	0.5955

7 Discussion and limitations

We acknowledge that this study is limited on several aspects: (a) people are self declaring to have an eating disorder, (b) within an online community, (c) expressing themselves according to the standard in use on Twitter and (d) we do not have any knowledge about their real life. However, we believe that they are representative of a part of people suffering from an eating disorder. A way to obtain more concrete results could be a joint study with clinical researchers in order to verify the validity of our study in a context outside the Internet and to improve it for other contexts. The annotation phase showed some limitations on the long run:

[10] https://radimrehurek.com/gensim/models/word2vec.html

- many human annotations fell under the GENERAL ED category found by CorEx: it could be possible to distinguish more topics that are a minority compared to others, but represent a big class altogether.
- Annotations on this domain, by human or by an algorithm, are complex: even though we decided only to use one label per sentence, we understand that there are some limitations, such as the co-presence of more topics in less than 20 words.

In the future, we would like to use both sentence similarities and a classification algorithm based on shallow parsing to capture them more accurately. We think that this could be improved by implementing syntactic rules (for example to capture implicit and explicit comparisons) and specific weights for words that are likely to be more in a category than in another. Take, for example, "burn"+"calorie" – we know that the word "calorie" is present in DIET, but the bigram "burn calorie" is likely to be more used in SPORT.

8 Conclusions

The main goals of this study were:

1. the creation of new resources, such as a pro-ED Twitter dataset and an annotated dataset both available under request – due to NDA, to facilitate and increase ED related studies on social media;
2. the exploration and sharing of alternative ways for the annotation experience, and the discovery of new keywords and textual items related to the studied issue, such as emojis, foreign language linguistic patterns, and uncommon use of words by employing two models: CorEx and WMD.

We believe that the work described in this paper can also be used in other online contexts where people's lives are in danger: suicide prevention, detection of depression signs, detection of harassment signs.

This work can be extended by using a classification framework to filter out dangerous expressions (encrypted or not), clustering them by topics, detecting keywords and increasing the number of keywords and topics. This will allow the early detection of possible online ED trends, such as the ABC diet and the Apple diet, or other dangerous online trends that were seen in the past (e.g., "*Blue Whale challenge*").

References

1 Alina Arseniev-Koehler, Hedwig Lee, Tyler McCormick, and Megan A Moreno. # proana: Pro-eating disorder socialization on Twitter. *Journal of Adolescent Health*, 58(6):659–664, 2016.

2 Carolina Figueras Bates. "I am a waste of breath, of space, of time" metaphors of self in a pro-anorexia group. *Qualitative Health Research*, 25(2):189–204, 2015.

3 David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

4 Leah Boepple and J Kevin Thompson. A content analytic comparison of fitspiration and thinspiration websites. *International Journal of Eating Disorders*, 49(1):98–101, 2016.

5 Dina LG Borzekowski, Summer Schenk, Jenny L Wilson, and Rebecka Peebles. e-ana and e-mia: A content analysis of pro–eating disorder web sites. *American journal of public health*, 100(8):1526–1534, 2010.

6 Patricia A Cavazos-Rehg, Melissa J Krauss, Shaina J Costello, Nina Kaiser, Elizabeth S Cahn, Ellen E Fitzsimmons-Craft, and Denise E Wilfley. "I just want to be skinny.": A content analysis of tweets expressing eating disorder symptoms. *PloS one*, 14(1):e0207506, 2019.

7 Stevie Chancellor, Yannis Kalantidis, Jessica A Pater, Munmun De Choudhury, and David A Shamma. Multimodal classification of moderated online pro-eating disorder content. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 3213–3226, 2017.

8 Stevie Chancellor, Zhiyuan Lin, Erica L Goodman, Stephanie Zerwas, and Munmun De Choudhury. Quantifying and predicting mental illness severity in online pro-eating disorder communities. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, pages 1171–1184, 2016.

9 Stevie Chancellor, Jessica Annette Pater, Trustin Clear, Eric Gilbert, and Munmun De Choudhury. # thyghgapp: Instagram content moderation and lexical variation in pro-eating disorder communities. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, pages 1201–1213, 2016.

10 Jonathan Chang, Sean Gerrish, Chong Wang, Jordan Boyd-graber, and David Blei. Reading tea leaves: How humans interpret topic models. In Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 22, pages 288–296. Curran Associates, Inc., 2009. URL: `https://proceedings.neurips.cc/paper/2009/file/f92586a25bb3145facd64ab20fd554ff-Paper.pdf`.

11 Yohan Chon, Yungeun Kim, Hyojeong Shin, and Hojung Cha. Topic modeling-based semantic annotation of place using personal behavior and environmental features. *Transportation*, 23:110, 2009.

12 Glen Coppersmith, Mark Dredze, Craig Harman, Kristy Hollingshead, and Margaret Mitchell. Clpsych 2015 shared task: Depression and PTSD on Twitter. In *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*, pages 31–39, 2015.

13 Kristen M Culbert, Sarah E Racine, and Kelly L Klump. Research review: What we have learned about the causes of eating disorders – a synthesis of sociocultural, psychological, and biological research. *Journal of Child Psychology and Psychiatry*, 56(11):1141–1164, 2015.

14 Munmun De Choudhury. Anorexia on tumblr: A characterization study. In *Proceedings of the 5th international conference on digital health 2015*, pages 43–50, 2015.

15 Elizabeth W Diemer, Julia D Grant, Melissa A Munn-Chernoff, David A Patterson, and Alexis E Duncan. Gender identity, sexual orientation, and eating-related pathology in a national sample of college students. *Journal of Adolescent Health*, 57(2):144–149, 2015.

16 Danielle A Gagne, Ann Von Holle, Kimberly A Brownley, Cristin D Runfola, Sara Hofmeier, Kateland E Branch, and Cynthia M Bulik. Eating disorder symptoms and weight and shape concerns in a large web-based convenience sample of women ages 50 and above: Results of the gender and body image (gabi) study. *International Journal of Eating Disorders*, 45(7):832–844, 2012.

17 Ryan J Gallagher, Kyle Reing, David Kale, and Greg Ver Steeg. Anchored correlation explanation: Topic modeling with minimal domain knowledge. *Transactions of the Association for Computational Linguistics*, 5:529–542, 2017.

18 Ryan J. Gallagher, Kyle Reing, David Kale, and Greg Ver Steeg. Anchored correlation explanation: Topic modeling with minimal domain knowledge. *Transactions of the Association for Computational Linguistics*, 5:529–542, 2017. `doi:10.1162/tacl_a_00078`.

19 Jannath Ghaznavi and Laramie D Taylor. Bones, body parts, and sex appeal: An analysis of# thinspiration images on popular social media. *Body image*, 14:54–61, 2015.

20 David Giles. Constructing identities in cyberspace: The case of eating disorders. *British journal of social psychology*, 45(3):463–477, 2006.

21 Grace J Johnson and Paul J Ambrose. Neo-tribes: The power and potential of online communities in health care. *Communications of the ACM*, 49(1):107–113, 2006.

22 Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. Fasttext.zip: Compressing text classification models. *arXiv preprint*, 2016. `arXiv:1612.03651`.

23 Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. *arXiv preprint*, 2016. arXiv:1607.01759.

24 Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *International conference on machine learning*, pages 957–966, 2015.

25 David E Losada, Fabio Crestani, and Javier Parapar. Overview of erisk: early risk prediction on the internet. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 343–361. Springer, 2018.

26 David E Losada, Fabio Crestani, and Javier Parapar. Overview of erisk 2019 early risk prediction on the internet. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 340–357. Springer, 2019.

27 David E Losada, Fabio Crestani, and Javier Parapar. erisk 2020: Self-harm and depression challenges. In *European Conference on Information Retrieval*, pages 557–563. Springer, 2020.

28 Luana Marques, Margarita Alegria, Anne E Becker, Chih-nan Chen, Angela Fang, Anne Chosak, and Juliana Belo Diniz. Comparative prevalence, correlates of impairment, and service utilization for eating disorders across us ethnic groups: Implications for reducing ethnic disparities in health care access for eating disorders. *International Journal of Eating Disorders*, 44(5):412–420, 2011.

29 Tomas Mikolov, Kai Chen, Greg S. Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013. arXiv:1301.3781.

30 Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL: https://proceedings.neurips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf.

31 David Mimno, Hanna M. Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. Optimizing semantic coherence in topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, page 262–272, USA, 2011. Association for Computational Linguistics.

32 Markus Moessner, Johannes Feldhege, Markus Wolf, and Stephanie Bauer. Analyzing big data in social media: Text and network analyses of an eating disorder forum. *International Journal of Eating Disorders*, 51(7):656–667, 2018.

33 Jessica A Pater, Brooke Farrington, Alycia Brown, Lauren E Reining, Tammy Toscos, and Elizabeth D Mynatt. Exploring indicators of digital self-harm with eating disorder patients: A case study. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW):1–26, 2019.

34 Danielle C Ransom, Jennifer G La Guardia, Erik Z Woody, and Jennifer L Boyd. Interpersonal interactions on online forums addressing eating concerns. *International Journal of Eating Disorders*, 43(2):161–170, 2010.

35 Azadeh A Rikani, Zia Choudhry, Adnan M Choudhry, Huma Ikram, Muhammad W Asghar, Dilkash Kajal, Abdul Waheed, and Nusrat J Mobassarah. A critique of the literature on etiology of eating disorders. *Annals of neurosciences*, 20(4):157, 2013.

36 Y. Rubner, C. Tomasi, and L. J. Guibas. A metric for distributions with applications to image databases. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pages 59–66, 1998. doi:10.1109/ICCV.1998.710701.

37 Yuanlong Shao, Yuan Zhou, Xiaofei He, Deng Cai, and Hujun Bao. Semi-supervised topic modeling for image annotation. In *Proceedings of the 17th ACM International Conference on Multimedia*, MM '09, page 521–524, New York, NY, USA, 2009. Association for Computing Machinery. doi:10.1145/1631272.1631346.

38 Shaina J Sowles, Monique McLeary, Allison Optican, Elizabeth Cahn, Melissa J Krauss, Ellen E Fitzsimmons-Craft, Denise E Wilfley, and Patricia A Cavazos-Rehg. A content analysis of an online pro-eating disorder community on reddit. *Body image*, 24:137–144, 2018.

39 Shabbir Syed-Abdul, Luis Fernandez-Luque, Wen-Shan Jian, Yu-Chuan Li, Steven Crain, Min-Huei Hsu, Yao-Chin Wang, Dorjsuren Khandregzen, Enkhzaya Chuluunbaatar, Phung Anh Nguyen, et al. Misleading health-related information promoted through video-based social media: anorexia on youtube. *Journal of medical Internet research*, 15(2):e30, 2013.

40 Catherine Victoria Talbot, Jeffrey Gavin, Tommy Van Steen, and Yvette Morey. A content analysis of thinspiration, fitspiration, and bonespiration imagery on social media. *Journal of eating disorders*, 5(1):1–8, 2017.

41 Marika Tiggemann, Owen Churches, Lewis Mitchell, and Zoe Brown. Tweeting weight loss: A comparison of# thinspiration and# fitspiration communities on Twitter. *Body Image*, 25:133–138, 2018.

42 Marcel Trotzek, Sven Koitka, and Christoph M Friedrich. Word embeddings and linguistic metadata at the clef 2018 tasks for early detection of depression and anorexia. In *CLEF (Working Notes)*, 2018.

43 Suppawong Tuarob, Line C Pouchard, Prasenjit Mitra, and C Lee Giles. A generalized topic modeling approach for automatic document annotation. *International Journal on Digital Libraries*, 16(2):111–128, 2015.

44 Greg Ver Steeg and Aram Galstyan. Discovering structure in high-dimensional data through correlation explanation. In *Advances in Neural Information Processing Systems*, pages 577–585, 2014.

45 Tao Wang, Markus Brede, Antonella Ianni, and Emmanouil Mentzakis. Detecting and characterizing eating-disorder communities on social media. In *Proceedings of the Tenth ACM International conference on web search and data mining*, pages 91–100, 2017.

46 Elad Yom-Tov, Luis Fernandez-Luque, Ingmar Weber, and Steven P Crain. Pro-anorexia and pro-recovery photo sharing: a tale of two warring tribes. *Journal of medical Internet research*, 14(6):e151, 2012.

47 Wei Zhang, Yan-Chuan Sim, Jian Su, and Chew-Lim Tan. Entity linking with effective acronym expansion, instance selection and topic modeling. In *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

48 Sicheng Zhou, Yunpeng Zhao, Rubina Rizvi, Jiang Bian, Ann F Haynos, and Rui Zhang. Analysis of Twitter to identify topics related to eating disorder symptoms. In *2019 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 1–4. IEEE, 2019.

A Twitter Corpus and Lexicon for Abusive Speech Detection in Serbian

Danka Jokić
University of Belgrade, Serbia

Ranka Stanković
Faculty of Mining and Geology, University of Belgrade, Serbia

Cvetana Krstev
Faculty of Philology, University of Belgrade, Serbia

Branislava Šandrih
Faculty of Philology, University of Belgrade, Serbia

Abstract

Abusive speech in social media, including profanities, derogatory and hate speech, has reached the level of a pandemic. A system that would be able to detect such texts could help in making the Internet and social media a better and more respectful virtual space. Research and commercial application in this area were so far focused mainly on the English language. This paper presents the work on building AbCoSER, the first corpus of abusive speech in Serbian. The corpus consists of 6,436 manually annotated tweets, out of which 1,416 were labelled as tweets using some kind of abusive speech. Those 1,416 tweets were further sub-classified, for instance to those using vulgar, hate speech, derogatory language, etc. In this paper, we explain the process of data acquisition, annotation, and corpus construction. We also discuss the results of an initial analysis of the annotation quality. Finally, we present an abusive speech lexicon structure and its enrichment with abusive triggers extracted from the AbCoSER dataset.

2012 ACM Subject Classification Computing methodologies → Natural language processing

Keywords and phrases abusive language, hate speech, Serbian, Twitter, lexicon, corpus

Digital Object Identifier 10.4230/OASIcs.LDK.2021.13

Funding Linked data development is supported by the COST Action CA18209-NexusLinguarum "European network for Web-centred linguistic data science". Access to SketchEngine and Lexonomy is provided by the ELEXIS project funded by the European Union's Horizon 2020 research and innovation program under grant number 731015.
Branislava Šandrih: COST Action CA18209-NexusLinguarum "European network for Web-centred linguistic data science"

Acknowledgements We would like to acknowledge the team of annotators that provided their time and efforts to help us build AbCoSER v1.0 corpus of abusive speech in Serbian.

1 Introduction

1.1 Motivation and research background

With the development of the Internet and the increasing use of online mass media and social networks, detection of inappropriate content and incitement to violence have gained importance. The concept of abusive speech, in the context of this paper, is an umbrella term for phenomena such as profanities, derogatory, and hate speech. One of the most cited definitions of hate speech comes from John T. Nockleby [44, 4], who perceives hate speech as "any communication that disparages a person or a group on the basis of some characteristic such as race, color, ethnicity, gender, sexual orientation, nationality, religion, or other characteristic".

According to a survey from 2014, 60% of Internet users witnessed name-calling, 25% saw that someone was physically threatened, and 24% noticed abuse over a long period [14]. According to more recent research from 2017, two-thirds of Americans stated that they had experienced some kind of harassment on the Internet [39]. Studies also show that 18% of children are involved in cyber-abuse, which leads to serious depression, and even suicide [12]. As far as Serbian law is concerned, any discrimination, endangering security, persecution, insults, and harassment on social networks are punishable [26, 4, 30]. Hate speech and flames are present in Serbian media and public discourse especially towards the LGBT population, Roma people, women and migrants [19].

Hate speech has become a major problem for all types of online platforms where an increasing amount of user-generated content appears: from comments on the web news portals, through social networks, to chats on real-time games [37]. Users are usually expected to report abusive speech, and then the site or social network moderators manually review the report. More advanced platforms use systems with regular expressions and "black" lists of words and expressions, to catch abusive language and remove posts [25]. There are also online portals such as HateBase.org that collect examples of online hate speech in all languages that can be used as trigger words for hate speech detection ([46, 41, 14, 13]). However, detecting hate speech by simply filtering by keywords is not a satisfactory solution, as interpretation can be influenced by the domain of the conversation, the context of the discourse, the objects that accompany the conversation (images, video, and audio materials), the time of publication and ongoing world events and the recipient of the message [38]. Given the huge amount of online material that is created every day, automatic methods are needed to detect and process this type of content.

One of the biggest problems that researchers have to solve before building the automatic hate speech detection systems is finding as many as possible publicly available annotated data sets of a considerable size, especially if the system will be based on deep learning [39]. Another problem researchers face is the non-existence of generally accepted definitions of hate speech and related phenomenon ([14, 38]), which leads to the use of different annotation schemes and categories definitions in various data set making it impossible to compare results of different systems [40]. An additional problem is that the available datasets usually focus on specific topics like misogyny or racist speech and do not cover all types of hate speech. In the last few years, hate speech has gained more attention from the research community, which led to the organization of several workshops, both independently or at international conferences that address problems of hate speech and related topics such as GermEval2018, Offenseval2019 and Offenseval 2020 ([47, 51, 52]).

Abusive language and its detection have also gained more attention recently. Casseli et al. [6] define abusive language as "hurtful language that a speaker uses to insult or offend another individual or a group of individuals based on their personal qualities, appearance, social status, opinions, statements, or actions. This might include hate speech, derogatory language, profanity, toxic comments, racist and sexist statements." From the definition itself, it is evident that abusive speech is a complex social and linguistic phenomenon [42]. Computational processing of such language requires the usage of finely-tuned, task-specific language tools and resources, especially for morphologically rich and low-resource languages such as Serbian. The main contribution of this work is the creation of the AbCoSER, the first abusive speech corpus in Serbian, that will, together with abusive speech lexicon, enable the development of automatic abusive speech detection systems for the Serbian language. In the course of this work, we leveraged existing annotation schemes and abusive term definitions as much as possible with the aim of creating a general data set convenient for the detection of a broad range of abusive topics. We already used this resource for the detection of abusive triggers and the augmentation of the abusive language lexicon.

1.2 Related work

In the past two decades, several methods and models for the detection of hate speech, abusive speech, toxic comments, and aggression on the Internet have been presented. From the natural language processing (NLP) perspective, the detection of hate speech can be viewed as a problem of classification: for a given statement, the system needs to determine whether it contains hate speech or not [36]. To achieve this goal systems usually apply text mining techniques. The majority of current hate speech, offensive, and abusive language detection systems in social media are based on lexicons or blacklists ([7, 10, 28, 34]). Their importance lies in the fact that a vast number of swear words and offences can be detected by using only lexicons. Razvan et al. [35], created an offensive word lexicon and then collected Twitter messages that contain at least one word from it. They concluded that the presence of a word in a tweet just indicates the possibility of offensive speech, and manual annotation is necessary to guarantee accurate tweets classification. The same lexicon was used in [48] to extract toxic conversations among adolescents on Twitter. While Pedersen [32] reported high accuracy of hate speech detection when using only a lexicon, the accuracy and F1 score were still lower compared to the state of the art [52] and the number of false positives was high, indicating that lexicons are not a sufficient resource for hate speech detection.

High-quality corpora of hate speech, offensive speech, and abusive language are very important as a first step in building an automated system for the detection of these phenomena ([51, 52, 1, 6]). Warner and Hirschberg [44] presented their research on hate speech toward minority groups in online text, with the main focus on anti-semitic language. Three annotators manually annotated a corpus of 1000 paragraphs taken from offensive websites and Yahoo user comments, with Fleiss kappa inter-annotator agreement at 0.63. Waseem and Hovy [45] created the renowned corpora of hate speech, consisting of 16,000 tweets grouped into 3 categories: racist (racism), sexist (sexism), neither. The corpus was built using bootstrapping, an iterative keyword search method. They created a decision-making list for the annotators, which helped them achieve an agreement score $\kappa = 0.84$ among the annotators. One of the most cited papers in this field was written by Nobata et al. [25]. They worked on several data sets consisting of comments from Yahoo Finance and Yahoo News pages. They performed an annotation experiment giving the same data set to trained users and untrained raters on Amazon's Mechanical Turk crowdsourcing platform and showed that better inter-annotator agreement was achieved by trained internal annotators. For binary categorization ("Clean" vs. "Abusive"), trained raters achieved an agreement rate 0.922 and Fleiss's Kappa 0.843 while Turkers agreement rate was 0.867 and Fleiss's Kappa 0.401. The agreement rate decreased when annotating abusive speech subcategories to 0.603 and 0.405 respectively. Davidson et al. [13] created a corpus of around 25,000 tweets with the idea of separating hate speech from other offensive speech. They used three categories to annotate the corpus: hate speech, just offensive speech, and neither using crowd-sourcing. The inter-annotator agreement score was 92%.

In the last few years, various data sets with multi-layered annotation appeared, which enabled both coarse and fine-grained classification. While Wiegand et al. [47] in the second layer classifies the type of insult (vulgar speech, insult, attack), Zampieri et al. [50] and Fisher et al. [15] emphasize the type and the target of offensive speech. The OLID data set and the scheme proposed by Zampieri et al. [50], used also at the SemEval2019 and SemEval 2020 competitions, gained popularity among researchers leading to the production of Turkish and Danish data sets that use this scheme ([11, 40]), as well as two new datasets, AbusEval and SWAD, which improve or use this data set to annotate a new one ([6, 27]). The multi-level universal annotation scheme, which includes the target of hatred or a type of

abusive speech, has many advantages. First of all, the classification can be done in several steps. Traditional machine learning can also be used at every step, which in the case of a smaller number of class examples gives better results than deep learning [31]. Another advantage is a simpler structure of the annotation decision tree, which can contribute to a better annotator agreement (the difference between the levels is clearer). The main advantage is that the same scheme can be used for general-purpose hate speech corpora, which includes several types of hate speech, and for specific corpora, which usually cover only one type of hate speech (racial hatred, misogyny, hatred of migrants, etc.).

The first system that dealt with hate speech detection in the Serbian language was described in [18]. The aim of this system was to detect newspaper articles that report on attacks and improper behaviour that are the result of national, racial, or religious hatred and intolerance. The system relied on electronic dictionaries of Serbian and local grammars that covered various patterns of hate speech and ways they were covered in newspaper articles. It should be noted that the focus of this research was different from hate speech detection today, as today's systems mostly deal with heath-speech directly (as found in user-generated content) and not with reports about it.

1.3 Paper outline

We describe in this paper the process of building the first data set of abusive language in Serbian. As the data source, we used tweets from the Twitter social network. Tweets from user timelines of 111 Twitter accounts were gathered and annotated by ten independent annotators working in pairs so that each tweet was annotated by two independent annotators while one supervising annotator resolved inconsistent annotations. Related work is given in Section 1.2, with a short overview of different approaches for developing an abusive speech data set. Our work in acquiring and annotating the data set, including a description of the annotation manual, is presented in Section 2. In Section 2.1, we describe the process of building the data-set of abusive language in Serbian. Further details about manual data annotation of corpus data are given in Section 2.2 while the extraction of abusive triggers is explained in Section 2.3. The Section 3 presents results of our research: Twitter data analysis (Subsection 3.1), the outcome of the annotation (Subsection 3.2) and the structure of the lexicon of abusive words (Subsection 3.3). We summarize the results of our research and indicate further research In discussion and conclusion Section 4.

2 Data Acquisition and Annotation

2.1 Data collection

When deciding which approach should we take when building the corpus of abusive language, several future implications were considered: 1) To the best of our knowledge, AbCoSER (Abusive Corpus for SERBIAN) is the first corpus tackling abusive language phenomenon in the Serbian language; 2) This corpus is to be used to enrich our lexicon of hate speech as described in [42]; 3) Classifiers trained on corpora containing general abusive speech, can be used to classify a domain hate speech corpus, while domain-specific classifiers perform poorly on the general data set and corpora from other hate speech domains ([46, 29]); therefore, instead of investigating domain-specific abusive speech, the phenomenon should be considered in a broader sense. Here we investigate abusive speech that covers vulgar speech, hate speech, and derogatory speech.

It is estimated that 2 to 3% of user-generated content contains abusive language. This means that the number of offensive messages is much smaller than non-offensive ones [38], which would impose practical problems on data annotators as well as automatic detection systems. To overcome this problem, researchers resort to searching by keywords or hashtags ([45, 50, 39, 36]), collecting comments directed to standard targets of abusive speech, or collecting comments from users who are notorious for using offensive language in their writing [47]. However, these approaches introduce bias into the data. While the keyword approach seems to be biased towards explicit expressions of abuse words used in the search [5], with the user timeline collection approach one must be careful when training the classifier so that it does not learn the writing style of a user instead of learning to detect the abusive messages [47]. In this work, the combined methodology is used with an iterative approach aiming at gathering as much abusive messages as possible. Twitter was used as a source for our data collection since it contains a much higher proportion of offensive language than other social networks [47]. Although a random sample of tweets would probably represent the unbiased set of data, we opted to sample tweets from the timeline of numerous Twitter user accounts. When sampling tweets from Twitter, we also imposed certain formal restrictions on the tweets to be extracted similar to those listed in [47]. They are as follows: 1) Each tweet had to be written in Serbian, 2) No tweet was allowed to contain any URLs, 3) No tweet was allowed to be a retweet.

At first, we started with the list of 80 user accounts gathered via crowd-sourcing. To this list, we added various users accounts whose tweets were reported as hate speech on the h8index,[1] an online platform for reporting hate speech, verbal violence, bullying, and discrimination on the Web. Initially, we gathered 450,000 tweets from the timeline of 120 user accounts via Twitter API[2] that were further cleaned by removing tweets that were retweeted from other users timeline and tweets containing URLs, leaving 150,000 tweets in the list. Although each tweet has a language column, in the majority of cases language of Serbian tweets was marked as *und* – unidentified – since Twitter cannot reliably recognize the Serbian language. For example, out of 150,000 tweets, only 8,000 were marked as tweets in Serbian, while 120,000 were marked as *und*. Therefore, we could not use this feature to filter tweets written in Serbian and have to rely on manual annotation.

In order to check how representative our data set is we sampled 200 tweets from it. Still, the ratio of tweets with abusive speech was just 12%. Therefore, the users' list was manually checked for the type of users and users were removed that are less likely to generate abusive speech such as:

1. Public users, like telecommunication and similar companies as well as newspapers and news portals were removed from the list of users since one cannot expect that this type of users will generate abusive speech, as proved in [11].
2. Fan pages and official pages of public persons, including politicians and sportsmen, or political parties were removed from the list for the same reason.
3. Users that tweet in a foreign language.
4. Users that do not generate abusive speech were detected by inspecting their timeline.

Thereafter, an initial list of a few seed words was identified, and Twitter was searched for occurrences of those words. We did not just add those tweets to our data set, we rather identified users that created those tweets and added them to the user list. The reason for

[1] https://h8index.org/
[2] https://developer.twitter.com/en/docs/twitter-api

such an approach was to retain the variety of offensive terms occurring in the collected tweets ([47, 6]). Finally, their followers and those who reply to abusive tweets were added to the list as well. At the end of this step, we extracted the timeline of 111 users, and we come up with 320,440 tweets. The next step was to remove duplicates, empty tweets, retweets, tweets with URLs, and tweets that contained just mentions. The list was reduced to 194,348 tweets. From this corpus, we randomly sampled 6,500 tweets. In the next step, we identified tweets written potentially in English by filtering out tweets whose language was marked with "en" (112 tweets). This set was manually checked and 64 English tweets were removed. The remaining 48 tweets were wrongly marked as written in English, while actually written in Serbian. The resulting data set had 6,436 tweets and this set was used for annotation.

Tweeter data differs significantly from other types of texts, e.g. books or newspaper articles, meaning that there are specific issues that have to be considered when processing such data. Some of them are:

1. Spelling, grammar and typing errors and regional variations are more frequent;
2. Frequent use of out of vocabulary words or intentionally misspelled words (e.g. fejv, lajna, QURAZ);
3. Excessive use of abbreviations, e.g. *nznm* (eng. I don't know), *mupm* (eng. go fuck your mum), *np* (eng. no problem), *jbt* (eng. fuck you), *jbg* (eng. fuck it) etc.;
4. Equal use of Cyrillic and Latin script, omission of diacritics, and different Unicode characters;
5. Use of foreign language words and emoticons (e.g. `:'-),:-P`, `:@)`);
6. Twitter-specific text: mentions, retweets and URLs as well as hashtags (e.g. #TLZP, #Utisak, #u6reci).

2.2 General Corpus annotation for classification of tweets

The reliability of the corpus annotation is key to the successful training of an automatic hate speech detection system [36]. Since a set of data annotated by only one person can be biased because it reflects his (or her) personal opinion [3], we decided each tweet be annotated by two independent annotators. Ten annotators participated in the annotation and therefore the data set was split into five parts and each part was annotated by an annotator pair. All annotators were native Serbian speakers.

To obtain a successful annotation scheme, it is important to satisfy the following criteria ([15, 11]): 1) The annotation scheme enables coarse and fine-grained classification; 2) The annotation scheme is accompanied by a detailed annotation guide; 3) The choice of classes corresponds to the expected use of data. Since our primary goal was to investigate abusive speech and its sub-categories, as well as to explore the possibility of distinguishing them, we decided to define a hierarchical two-layer annotation scheme and categories, which is similar to one developed by Nobata et al. [25] and in adherence to the annotation guidelines resulted from the research of Fortuna et al. [16]. The annotation was performed considering the content of a whole tweet, as was the case for the majority of data sets ([47, 23, 50, 49, 36]). At the first level annotators marked a tweet as abusive (TRUE) or non-abusive (FALSE). At the second level annotators determined the category of abusive speech in tweets marked as abusive:

1. Profanity (PROF), the tweet contains simplicity and vulgarity (e.g. "lakše se kenja i preti iz anonimnosti..."/"it's easier to talk shit and threaten out of anonymity ...");
2. Hate speech (HS), if a tweet contains an attack, disparagement, or promotion of hatred towards a group of people or members of that group on the basis of race, ethnicity, nationality, gender, religion, political orientation, sexual orientation (e.g. "@USER Što se mene tiče ne trebate nam. Iz Crne Gore dolaze mafijaški klanovi. Nismo mi vama poslali

TWEET DATA			LEVEL 1	LEVEL 2			LANG
Index	Tweet text	Lang	Abusive	Vulgar	Hate	Derogatory	Correction
138451	@Stefan_Visoki Ali bitno da je nacija poruku shvatila nedvosmisleno: "oni nas mnogo jako jebu u mozak"	und	☑	☑	☐	☑	sr
82461	O miševima i mudima.	und	☑	☑	☐	☐	sr
176861	ali “digla dzevu u zari” idi bre kuvaj neki rucak nesto tako	und	☑	☐	☑	☐	sr
16907	Na koji broj se šalje SMS za lečenje Marijana Rističevića?	und	☑	☐	☐	☑	sr
153622	@Milan92551954 Рада Ђурић је говно, било где да ради, чак и у медију власти, остаће безлично говно.	sr	☑	☑	☐	☑	sr
230346	"Jbte, zarazio me" -Jbte, dopustio si. I to vam je cela priča, jer verovati nekom na reč odavno ne pije vodu..	sl	☑	☑	☐	☐	sr
256251	@KalasturaB Jesi kalaštura... Mrš od dece.	und	☑	☑	☐	☑	sr
2154	iz komentara sam zaključio da je čovek poljak jer često piše na poljskom, a moja izvanredna moć dedukcije zaključuje da je čovek serviser mašina, jer naravno da niko ne poseduje na desetine starih modela veš mašina	und	☐	☐	☐	☐	sr
90923	Да сам поднаслов књиге био бих "Догодовштине једног пушача на почетку трећег миленијума".	sr	☐	☐	☐	☐	sr

Figure 1 The annotation interface with examples.

mafijaše. Da nije Srba u Crnoj Gori ta država bi mi bila draga koliko i Hrvatska.” / “@USER As far as I’m concerned we don’t need you. Mafia clans come from Montenegro. We didn’t send you mobsters. If there were no Serbs in Montenegro, that country would be as dear to me as Croatia.”);

3. Derogatory speech (DS), a tweet is used to attack or humiliate an individual or group in a general sense, not in a way hate speech does (e.g. “@USER To je jedna budala, ne veruj mu ništa.”/“@USER He’s a fool, don’t believe him.”)
4. Other (OTH), abusive speech that does not belong to the above-mentioned categories e.g. ironic or sarcastic tweets.

An abusive tweet belongs to at least one of the categories from the second annotation level. An example of a tweet that belongs to both PROF and DR category is “@USER NAME je govno, bilo gde da radi, čak i u mediju vlasti, ostaće bezlično govno” (eng. “NAME is shit, wherever he/she works, even in the government media, he/she will remain an impersonal shit”).

All annotators were provided with training and annotation guidelines containing examples similar to ([33, 25]). For each of the category, annotators obtained its definition, some examples and an indicative list of trigger words characteristic to it, as described in [42]. Besides annotation guidelines, annotators received the decision list for abusive speech identification similar to the one used in [45], but upgraded and adopted for the general abusive speech. Since Twitter does not identify Serbian as a language successfully, and thus the language column of a tweet could not be relied upon, the annotators were given one more task – to check the language of a tweet and whether it could be interpreted. They needed to mark tweets with meaningless content, tweets written in a foreign language or multilingual tweets. After annotating the initial set of 200 tweets, an additional workshop with annotators was conducted to comment on the first annotation results and discuss discovered problems. Annotation was done online using Google sheets,as presented in Figure 1.

As annotation guidelines in the form of the decision tree are proven to be good a practice ([45, 6, 23]), we prepared the guidelines for annotators in the same format (shown in Figure 2). One can see from the decision tree that a tweet marked as abusive has to be tested for each subcategory, since, as mentioned earlier, one tweet can belong to one or more subcategories. Annotators had a tab in their annotation interface with examples of all possible annotation combinations.

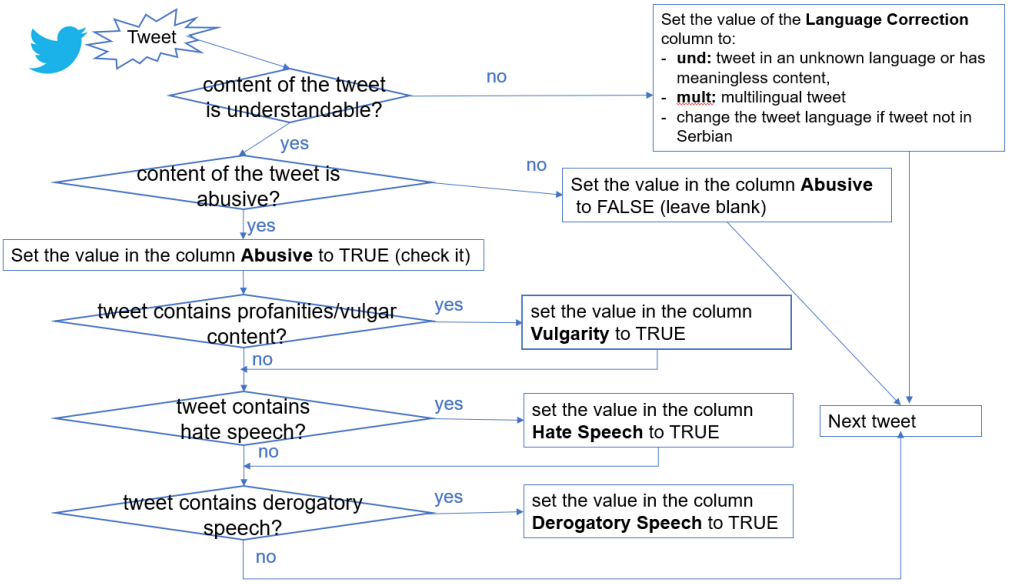


Figure 2 Annotation guidelines in the form of the decision tree.

Tweet index	Tweet text	Abusive trigger (lemma)	Abusive class	Score
169879	E vala ja bih prišao da je udavim odmah, da se više ne bori za dah. Majku li vam jebem američku!!!!!!! (eng. Well, I would approach her to drown her right away, so she doesn't fight for her breath anymore. Fuck your American mother !!!!!!!)	udaviti (eng. drown)	Threat	4
169879		jabati (eng, fuck)	Offensive	4

Figure 3 Abusive span annotation entries for one tweet example.

2.3 Annotation of abusive token spans for lexicon building

Several systems for the detection of abusive language as well as data sets were developed for English ([50, 13, 49, 45]), German [46], Slovene [23], Danish [40] and Turkish [11] that classify whole documents or comments/tweets as abusive without identifying which token spans were abusive. It would be very useful to have those abusive triggers highlighted so that human moderators can react timely to the abusive content. Following the Toxic Spans Detection task on Semeval 2021, and in line with our goal to enrich our lexicon of abusive speech with new entries and the usage examples [42], the additional manual annotation was conducted on 1,564 tweets that at least one of the annotators marked as abusive. This set of tweets was divided and shared among the already trained annotators with a task to detect triggers in each of the tweets. The annotators were given oral and written instructions together with examples of abusive speech categories and respective triggers as discussed in [42]. Figure 3 presents an abusive tweet in which two triggers were identified, classified into different categories of abusiveness and assigned abusiveness score. The purpose of identifying abusive triggers is to use them to enrich the lexicon of abusive words whose structure is presented in Section 3.3.

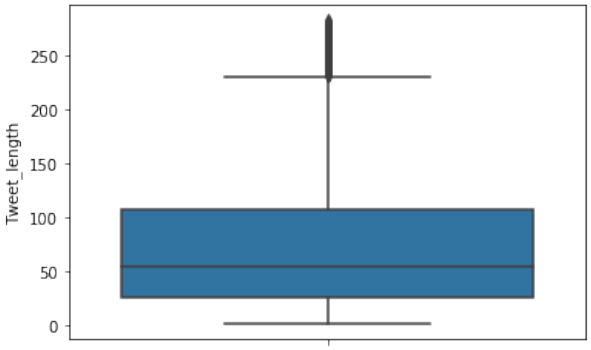


Figure 4 The distribution of tweets length.

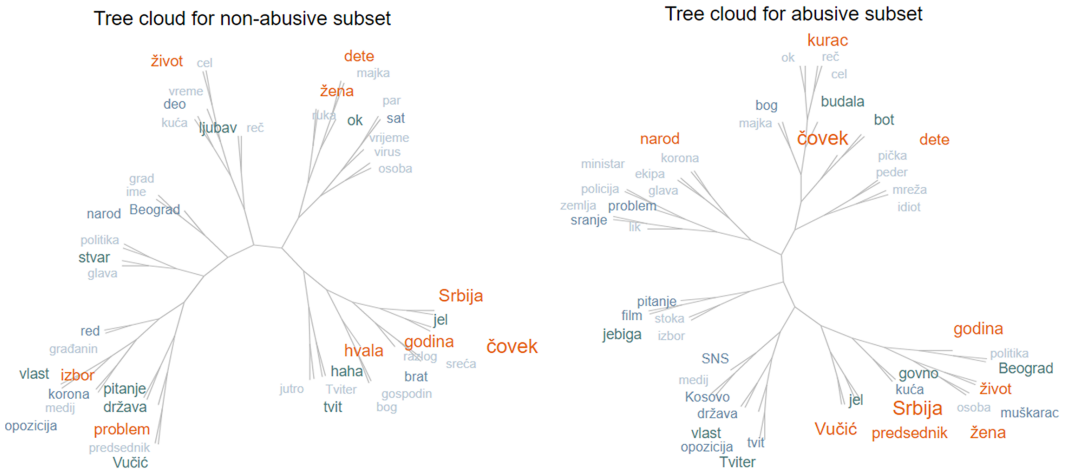


Figure 5 Tree clouds of abusive language corpus: the non-abusive subset (left) and abusive subset (right).

3 A Corpus and Lexicon for Abusive Speech

3.1 Analysis of the twitter corpus

The distribution of our corpus tweets length after removal of mentions is shown in Figure 4: the median value is 54 characters and the mean value is 78.56. As explained in Subsection 2.1, the corpus of tweets needs further pre-processing. First, all Cyrillic characters were replaced with corresponding Latin script characters[3], then punctuation, special and non-printable Unicode characters were also removed, and hash sign # deleted from hashtags. In the end, Tweet tokenizer from Python nltk[4] tokenizes the tweets removing at the same time mentions and an excessive number of repeated characters in tokens.

After data pre-processing, the data visualization technique is applied to cleaned tweets corpus to gain insights into data content. The tree cloud model [17] was employed for data visualization of non-abusive and abusive subsets of data (Figure 5). Besides depicting more frequent words in a larger font, words are also arranged on a tree in a way that reflects their

[3] cyrtransli Python library is available at `https://pypi.org/project/cyrtranslit/`.
[4] `https://www.nltk.org` Python Natural Language Toolkit

Table 1 The inter-annotator agreement per categories of abusive speech.

Category/Subcategory of hate speech	The inter-annotator agreement, accuracy
Offensive/Non-offensive	$\kappa = 0.513$, $accuracy = 0.860$
Profanities	$\kappa = 0.612$, $accuracy = 0.956$
Hate speech	$\kappa = 0.263$, $accuracy = 0.949$
Derogatory speech	$\kappa = 0.370$, $accuracy = 0.895$

semantic proximity in the text. We are interested in the most common words, as well as hashtags, in the data sets for both labels (abusive and non-abusive). We can notice that the most frequently used words in the non-abusive dataset are rather neutral words such as *hvala* (eng. thank you), *ljudi* (eng. people), *godina* (eng. year), *problem* (eng. problem), *pitanje* (eng. question), etc. Word *korona* appears among the top 15 words, which is due to the current Covid-19 pandemic. We assumed that our data set might be biased considering the period of data capturing and that proved to be true. In the non-abusive set, many high-frequency words referring to Serbian authority and state politics occur. No abusive word was identified among the top 50 words of this subset.

On the other hand, when we looked at the top 50 words in the abusive speech subset, we noticed that it contained a number of derogatory and vulgar words such as different forms of *jebati* (to fuck), *peder* (gay) *kurac* (dick), *budala* (fool), *govna* (shit), etc. High on the list of the most frequent words are also *Srbija* (Serbia), *sns* (the acronym of *Srpska Napredna Stranka*, Serbian ruling political party), and *Vučić* (the president of Serbia). Among the top 15 words on the list is also *žena* (woman).

We also made a hypothesis that hashtags can be indicators of abusive tweets. Therefore we looked at hashtags used in non-abusive and abusive tweets to check whether there is some pattern of their usage. Two lists of hashtags were created from the raw tweet data and compared with the frequency of hashtag appearance. The distributions of hashtags of non-abusive and abusive tweets were analysed, but we could not confirm our hypothesis because the number of tweets with hashtags was rather low – only 110 non-abusive and 24 abusive tweets contained hashtags.

3.2 Statistics and availability of the corpus

As a measure of inter-annotator agreement, we used Cohen's Kappa coefficient. The results for each of the categories are presented in Table 1. Cohen's Kappa score for the binary annotation Offensive/Non-offensive speech equals 0.513. When further analyzing the results, we noted that the best agreement was achieved with the annotation of profanities ($kappa = 0.612$), while the worst results were for the hate speech ($kappa = 0.263$). Since this level of agreement was not satisfactory, one of the authors of this paper acted as the 3rd supervising annotator, whose task was to resolve annotations on which the first round annotators disagreed. In total, 2,185 differences were identified and harmonized at both annotation levels and the decision was made for all of them.

The resulting data set has in total 1,416 tweets labelled as abusive, out of a total of 6,436 tweets in the data set. 472 tweets are marked as PROF, 273 as HS, 843 as DS, and 169 as OTH. 637 tweets are assigned to more than one abusive category. We are currently working on expanding the data set with additional tweets after which it will be made publicly available.

3.3 The lexicon of abusive speech

The lexicon of abusive speech, consisting of words that could be used as triggers for the recognition of abusive language is being built, with the idea that the Serbian system for the recognition and normalization of abusive expressions will also take into consideration phrases and figurative speech as indicators.

In addition to the improved version of Hurtlex [2], resources that can be useful for the creation of a lexicon of offensive words are lists of swear words, curses, abusive expressions, existing general dictionaries, slang dictionaries, surveys and contributions through crowd-sourcing, translation of dictionaries and lexicons from other languages, lexicons of sentiment words and expressions, rhetorical figures, etc. To expand the dictionary, synsets from the Serbian WordNet and the dictionary of synonyms will be used for linking with Twitter examples.

Regarding the categorization of terms in the lexicon, the Hatebase scheme[5] was used as a guideline because it is already a kind of a standard in this area, and then supplemented with additional categories according to hate targets as presented in [41], namely *Category* can be Race, Behavior, Physical, Sexual orientation, Class, Gender, Ethnicity, Disability, Religion, Other. A certain term in the lexicon can be assigned several categories, in case it appears in the context of several types of hate speech. The *Severity* attribute has values within a range 1–5 that represent the degree of insult that can be assigned or automatically calculated from the annotated data set presented in Section 2.3. The value *OffensLevel* is assigned as a measure of a chance that the word is used in the offensive meaning and it is calculated based on the number of different meanings in the comprehensive explanatory dictionary of Serbian, and need to match neither corpus nor probability of use. An excerpt from the dictionary for the word *lopov* (thief) is presented in Listing 1. It can be seen that this word can be used to refer to immoral or criminal activities or as a derogatory word to insult someone.

Information integration beyond the level of dictionaries and across the language resource community has become an important concern. The most promising technology for information integration is the Linked (Open) Data (LOD) paradigm that is used for publishing lexical resources by using URIs to unambiguously identify lexical entries, their components and their relations in the web of data. Moreover, it is used to make lexical data sets accessible via http(s), to publish them in accordance with W3C-standards such as RDF and SPARQL, and to provide links between lexical data sets and other LOD resources [8].

The goal of our research is to make its results compatible with the Linked Data approach, using its set of design principles for sharing machine-readable interlinked data on the Web. This vision of globally accessible and linked data on the internet is based on RDF standards for semantic web, using RDF serialisation for data representation. To that end, our approach envisages export of trigger words as lexical data in RDF that is compliant with the *The OntoLex Lemon Lexicography Module*[6], lexicog [5], as an extension of Lexicon Model for Ontologies (lemon)[7] [24]. This is also in line with activities within NexusLinguarum COST action[8], which promotes synergies across Europe between linguists, computer scientists, terminologists, language professionals, and other stakeholders in industry and society in

[5] `https://hatebase.org/` The world's largest structured repository of regionalized, multilingual hate speech
[6] `https://www.w3.org/2019/09/lexicog/`
[7] `https://www.w3.org/2016/05/ontolex/`
[8] `https://nexuslinguarum.eu/`

Listing 1 An excerpt from the XML version of the dictionary.

```
<LexicalEntry id="SR0001" lng="sr" pos="n" Probability="0.8">
  <lemma>lopov</lemma>
  <OffensCategories>
    <OffensCategory Severity="4" OffensLevel="0.7">
      <Examples type="Immoral or criminal activities">
        <Example beginIndex="0" endIndex="6" form="lopovu">
                 lopovu je mesto u zatvoru</Example>
        <Example beginIndex="19" endIndex="25" form="lopovi">
                 svi političari su lopovi</Example>
      </Examples>
    </OffensCategory>
    <OffensCategory Severity="3" OffensLevel="0.4">
      <Examples type="Derogatory words and insults">
        <Example beginIndex="12" endIndex="17" form="lopov"
             type="MWU">ružan kao lopov</Example>
      </Examples>
    </OffensCategory>
  </OffensCategories>
</LexicalEntry>
```

Listing 2 An excerpt from RDF version of dictionary.

```
:lopov a ontolex:lopov;
   dct:language <http://id.loc.gov/vocabulary/iso639-1/sr> ;
   lexinfo:partOfSpeech lexinfo:noun;
   ontolex:lexicalForm :lopov-form;
   ontolex:sense :lopov-sense.
:lopov-form a ontolex:Form;
   ontolex:writtenRep "lopov"@sr.
:lopov-sense skos:definition "onaj koji krade, kradljivac,
   lupež; otimač, pljačkaš; prevarant, lupež"@sr;
   ontolex:reference <https://www.wikidata.org/wiki/Q3562775>.
```

order to investigate and extend the area of linguistic data science. As an illustration, the RDF model in Turtle syntax[9] is presented in Listing 2, using the same word *lopov* (thief) as an example.

In addition, the usage of the novel module for frequency, attestation and corpus information for Ontolex Lemon (FrAC) [9] is developed. Our intention is to select trigger words that can be found in the corpus AbCoSER and to link usage samples to actual tweets. Lexical variants of trigger words were also included, which is especially important in this case because Twitter users tend to use many irregular forms. Since Serbian is a highly inflective and morphologically rich language that uses a lot of different word suffixes to express different grammatical, syntactic, or semantic features, we also established the relation with the Serbian electronic dictionaries and the management platform Leximirka (Figure 6) [22], which enables the recognition of all inflected forms of trigger words.

For the ranking and selection of illustrative tweets (or its parts) as a kind of dictionary usage examples, we have used a weighted score derived from lexical, word-based and other features (e.g., sentence length, number of all no space chars, digits, weird chars, commas, full

[9] https://www.w3.org/TR/turtle/

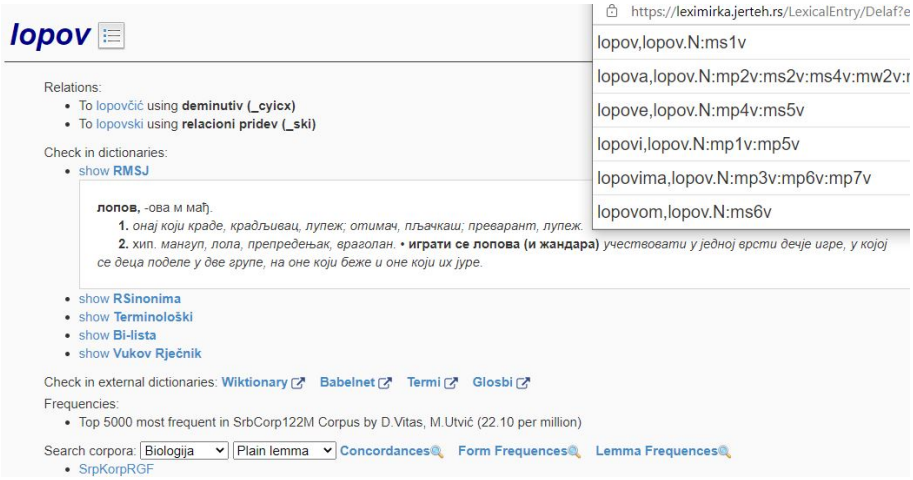


Figure 6 The LEXIMIRKA application for lexical database management and use.

stops, punctuation, number of all tokens, average token length, max token length). We use this score to rank examples, but system allows a different number of examples for a different purpose. For example, for dict2wec [43] a larger number of examples will be provided.

The relative frequency (normalized per million) was assigned to lexical entries both for the abusive language (derived from the abusive tweet corpus) and for neutral language (derived from the corpus of non-abusive tweets), which enables calculation of the so-called keyness score, which should represent the extent of the frequency difference. These frequencies can also be compared with the corpus of standard Serbian (as reference). Since frequency information is a crucial component in human language technology, the FrAC module facilitates sharing and utilising this valued information [9], as presented in Listing 3.

4 Discussion and conclusion

In this paper, we presented AbCoSER 1.0, the first corpus of abusive language in Serbian which consists of tweets. We explained the process of data acquisition, annotation, and corpus construction. All tweets were annotated by two independent annotators, but as explained in Section 3.2, the inter-annotator agreement was moderate. Possible causes might be: 1) Lack of the generally accepted definitions of abusive speech ([14, 38]), it is often necessary to consider tweets on a case-by-case basis, 2) Individual bias of annotators due to cultural differences, personal sensibility and/or knowledge of the phenomenon, 3) Vague or incomplete annotation instructions, 4) Overlapping of abusive speech sub-categories. In general, our results are in alignment with the findings of other researchers who reported low inter-annotator agreement scores ([21, 33, 23]) As Ross et al. [36] noted, hate speech is a very vague concept that requires better definitions and guidelines. One of the characteristics of our annotation scheme is that tweets containing swear words corresponding to the category of Profanity in our data set can also be used in non-abusive informal speech. Moreover, they are often used to emphasize a positive phenomenon as in the example *Tajson je i sa 54g jebena mašina...* (Tyson is at 54 still a fucking machine...) and not just in the context of insults. The instruction to mark even those tweets as abusive might cause cognitive dissonance with annotators since they would in a regular case mark it as non-abusive ([27, 6]). This annotation approach was chosen to facilitate automatic detection of abusive speech by a system based on machine learning techniques. There are also cases when negation

Listing 3 An excerpt from RDF version with frequency and attestations.

```
# subproperty definition for frequency in twitter corpus
:atvitkoFrequency rdfs:subClassOf frac:CorpusFrequency .
:atvitkoFrequency rdfs:subClassOf [
   a owl:Restriction ;
   owl:onProperty frac:corpus ;
   owl:hasValue <https://app.sketchengine.eu/#
      dashboard?corpname=user%2Franka%2Fatwitco>] .
# frequency assessment (in twitter corpus)
:lopov frac:frequency [
   a :atvitkoFrequency;
   rdf:value "17"^^xsd:int].
# usage examples as attestations
:lopov frac:attestation attestation_1324567;
attestation_1324567 a frac:Attestation ;
   cito:hasCitedEntity   <https://app.sketchengine.eu/#
      dashboard?corpname=user%2Franka%2Fatwitco> ;
   rdfs:comment "Immoral or criminal activities" ;
   frac:locus :locus_2415677;
   frac:quotation "svi političari su lopovi." .
:locus_2415677 a :Occurrence ;
   nif:beginIndex 19 ;
   nif:endIndex 25.
```

and emoticons change the meaning, usage of irony and sarcasm that is hard to detect in written language, as well as the necessity to possess knowledge about the world and current circumstances to understand and annotate the message.

In the next phase, we plan to extend the AbCoSER corpus with new tweets and with texts from other sources e.g. online news comments. Meanwhile, we started developing models for the automatic classification of abusive tweets and the first results are comparable with the results on similar data sets for other languages ([25, 44, 47]). The focus of our current research is the usage of a hybrid approach that combines machine learning and lexical resources. Finally, a user-friendly interface that will enable the use of these resources on the Web is under development. As for the development of the lexical resources, we plan to prepare an ontology for the classification of abusive data, including tweets, to tackle ambiguity in hate speech detection [20]. The development of the lexicon of abusive words and the ontology using VocBench[10] will continue. We also plan to enrich the lexicon with triggers identified during the annotation of abusive token spans as described in Section 2.3 and use it to upgrade the AbCoSER corpus.

References

1 Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, 2019.

2 Elisa Bassignana, Valerio Basile, and Viviana Patti. Hurtlex: A multilingual lexicon of words to hurt. In *5th Italian Conference on Computational Linguistics, CLiC-it 2018*, volume 2253, pages 1–6. CEUR-WS, 2018.

[10] `http://vocbench.uniroma2.it`

3 Bastian Birkeneder, Jelena Mitrovic, Julia Niemeier, Leon Teubert, and Siegfried Handschuh. upInf-Offensive Language Detection in German Tweets. In *Proceedings of the GermEval 2018 Workshop*, pages 71–78, 2018.

4 Andrej Blagojević et al. The normative framework of hate speech in Serbia and Serbian media. *FACTA UNIVERSITATIS-Law and Politics*, 14(1):81–95, 2016.

5 Julia Bosque-Gil, Jorge Gracia, and Elena Montiel-Ponsoda. Towards a Module for Lexicography in OntoLex. In *Proceedings of the LDK workshops: OntoLex, LDK 2017, Galway, Ireland*, volume 1899, pages 74–84, 2017.

6 Tommaso Caselli, Valerio Basile, Jelena Mitrović, Inga Kartoziya, and Michael Granitzer. I Feel Offended, Don't Be Abusive! Implicit/Explicit Messages in Offensive and Abusive Language. In Calzolari et al., editor, *Proceedings of the Twelfth International Conference on Language Resources and Evaluation (LREC 2020)*, Marseille, France, May 11–16 2020. European Language Resources Association (ELRA).

7 Ying Chen, Yilu Zhou, Sencun Zhu, and Heng Xu. Detecting offensive language in social media to protect adolescent online safety. In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, pages 71–80. IEEE, 2012.

8 Christian Chiarcos, Christian Fäth, and Maxim Ionov. The ACoLi dictionary graph. In *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, pages 3281–3290, Marseille, France, 2020. European Language Resources Association. URL: `https://www.aclweb.org/anthology/2020.lrec-1.401.pdf`.

9 Christian Chiarcos, Maxim Ionov, Jesse de Does, Katrien Depuydt, Fahad Khan, Sander Stolk, Thierry Declerck, and John Philip McCrae. Modelling Frequency and Attestations for OntoLex-Lemon. In *Proceedings of the 2020 Globalex Workshop on Linked Lexicography*, pages 1–9, Marseille, France, 2020. European Language Resources Association. URL: `https://www.aclweb.org/anthology/2020.globalex-1.1.pdf`.

10 Davide Colla, Caselli Tommaso, Valerio Basile, Jelena Mitrović, and Granitzer Michael. GruPaTo at SemEval-2020 Task 12: Retraining mBERT on Social Media and Fine-tuned Offensive Language Models. In *Proceedings of the 14th International Workshop on Semantic Evaluation(SemEvaleval)*, 2020.

11 Çağrı Çöltekin. A corpus of Turkish offensive language on social media. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6174–6184, 2020.

12 Maral Dadvar, Dolf Trieschnigg, Roeland Ordelman, and Franciska de Jong. Improving cyberbullying detection with user context. In *European Conference on Information Retrieval*, pages 693–696. Springer, 2013.

13 Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11/1, 2017.

14 Mai ElSherief, Vivek Kulkarni, Dana Nguyen, William Yang Wang, and Elizabeth Belding. Hate lingo: A target-based linguistic analysis of hate speech in social media. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 12/1, 2018.

15 Darja Fišer, Tomaž Erjavec, and Nikola Ljubešić. Legal framework, dataset and annotation schema for socially unacceptable online discourse practices in Slovene. In *Proceedings of the first workshop on abusive language online*, pages 46–51, 2017.

16 Paula Fortuna, Juan Soler, and Leo Wanner. Toxic, hateful, offensive or abusive? what are we really classifying? an empirical analysis of hate speech datasets. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6786–6794, 2020.

17 Philippe Gambette and Jean Véronis. Visualising a text with a tree cloud. In *Classification as a Tool for Research*, pages 561–569. Springer, 2010.

18 Cvetana Krstev, Sandra Gucul, Duško Vitas, and Vanja Radulović. Can we make the bell ring? In *Proceedings of the Workshop on a Common Natural Language Processing Paradigm for Balkan Languages*, pages 15–22, 2007.

19 Ivana Krstić. Report on the use of hate speech in Serbian media, 2020. URL: https://rm.coe.int/hf25-hate-speech-serbian-media-eng/1680a2278e.

20 K. Kumaresan and K. Vidanage. Hatesense: Tackling ambiguity in hate speech detection. In *2019 National Information Technology Conference (NITC)*, pages 20–26, 2019. doi:10.1109/NITC48475.2019.9114528.

21 Irene Kwok and Yuzhou Wang. Locate the hate: Detecting tweets against blacks. In *Proceedings of the twenty-seventh AAAI conference on artificial intelligence*, pages 1621–1622, 2013.

22 Biljana Lazić and Mihailo Škorić. From DELA based dictionary to Leximirka lexical database. *Infotheca – Journal for Digital Humanities*, 19(2):81–98, 2020. doi:10.18485/infotheca.2019.19.2.4.

23 Nikola Ljubešić, Darja Fišer, and Tomaž Erjavec. The FRENK datasets of socially unacceptable discourse in Slovene and English. In *International Conference on Text, Speech, and Dialogue*, pages 103–114. Springer, 2019.

24 John McCrae, Guadalupe Aguado-de Cea, Paul Buitelaar, Philipp Cimiano, Thierry Declerck, Asunción Gómez-Pérez, Jorge Gracia, Laura Hollink, Elena Montiel-Ponsoda, Dennis Spohr, et al. Interchanging lexical resources on the Semantic Web. *Language Resources and Evaluation*, 46(4):701–719, 2012. doi:10.1007/s10579-012-9182-3.

25 Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. Abusive language detection in online user content. In *Proceedings of the 25th international conference on world wide web*, pages 145–153, 2016.

26 Government of the Republic of Serbia. Criminal code of the Republic of Serbia. *Službeni glasnik*, 35:1–104, 2019.

27 Endang Wahyu Pamungkas, Valerio Basile, and Viviana Patti. Do you really want to hurt me? predicting abusive swearing in social media. In *The 12th Language Resources and Evaluation Conference*, pages 6237–6246. European Language Resources Association, 2020.

28 Endang Wahyu Pamungkas, Valerio Basile, and Viviana Patti. Misogyny detection in twitter: a multilingual and cross-domain study. *Information Processing & Management*, 57(6):102360, 2020.

29 Endang Wahyu Pamungkas and Viviana Patti. Cross-domain and cross-lingual abusive language detection: A hybrid approach with deep learning and a multilingual lexicon. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 363–370, 2019.

30 Nikola Pantelić. CRIMINAL OFFENSES COMMITTED ON SOCIAL NETWORKS: Structure of the offense and position of the perpetrator, 2017.
URL: https://www.paragraf.rs/100pitanja/krivicno_pravo/krivicna-dela-izvrsena-na-drustvenim-mrezama- struktura-dela-i-polozaj-izvrsioca.html.

31 Ji Ho Park and Pascale Fung. One-step and two-step classification for abusive language detection on twitter. *arXiv preprint*, 2017. arXiv:1706.01206.

32 Ted Pedersen. Duluth at SemEval-2019 task 6: Lexical approaches to identify and categorize offensive tweets. *arXiv preprint*, 2020. arXiv:2007.12949.

33 Fabio Poletto, Marco Stranisci, Manuela Sanguinetti, Viviana Patti, and Cristina Bosco. Hate speech annotation: Analysis of an Italian twitter corpus. In *4th Italian Conference on Computational Linguistics, CLiC-it 2017*, volume 2006, pages 1–6. CEUR-WS, 2017.

34 Amir H Razavi, Diana Inkpen, Sasha Uritsky, and Stan Matwin. Offensive language detection using multi-level classification. In *Canadian Conference on Artificial Intelligence*, pages 16–27. Springer, 2010.

35 Mohammadreza Rezvan, Saeedeh Shekarpour, Lakshika Balasuriya, Krishnaprasad Thirunarayan, Valerie L Shalin, and Amit Sheth. A quality type-aware annotated corpus and lexicon for harassment research. In *Proceedings of the 10th ACM Conference on Web Science*, pages 33–36, 2018.

36 Björn Ross, Michael Rist, Guillermo Carbonell, Benjamin Cabrera, Nils Kurowsky, and Michael Wojatzki. Measuring the reliability of hate speech annotations: The case of the european refugee crisis. *arXiv preprint*, 2017. `arXiv:1701.08118`.

37 Haji Mohammad Saleem, Kelly P Dillon, Susan Benesch, and Derek Ruths. A web of hate: Tackling hateful speech in online social spaces. *arXiv preprint*, 2017. `arXiv:1709.10159`.

38 Anna Schmidt and Michael Wiegand. A survey on hate speech detection using natural language processing. In *Proceedings of the fifth international workshop on natural language processing for social media*, pages 1–10, 2017.

39 Alessandro Seganti, Helena Sobol, Iryna Orlova, Hannam Kim, Jakub Staniszewski, Tymoteusz Krumholc, and Krystian Koziel. NLPR@ SRPOL at SemEval-2019 Task 6 and Task 5: Linguistically enhanced deep learning offensive sentence classifier. *arXiv preprint*, 2019. `arXiv:1904.05152`.

40 Gudbjartur Ingi Sigurbergsson and Leon Derczynski. Offensive language and hate speech detection for Danish. *arXiv preprint*, 2019. `arXiv:1908.04531`.

41 Leandro Silva, Mainack Mondal, Denzil Correa, Fabrício Benevenuto, and Ingmar Weber. Analyzing the targets of hate in online social media. *arXiv preprint*, 2016. `arXiv:1603.07709`.

42 Ranka Stanković, Jelena Mitrović, Danka Jokić, and Cvetana Krstev. Multi-word Expressions for Abusive Speech Detection in Serbian. In *Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons*, pages 74–84, 2020.

43 Julien Tissier, Christophe Gravier, and Amaury Habrard. Dict2vec: Learning word embeddings using lexical dictionaries. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 254–263, 2017.

44 William Warner and Julia Hirschberg. Detecting hate speech on the world wide web. In *Proceedings of the second workshop on language in social media*, pages 19–26, 2012.

45 Zeerak Waseem and Dirk Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93, 2016.

46 Michael Wiegand, Josef Ruppenhofer, Anna Schmidt, and Clayton Greenberg. Inducing a lexicon of abusive words – a feature-based approach. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, June 1–June 6, 2018, New Orleans, Louisiana, Vol. 1*, 2018.

47 Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. Overview of the germeval 2018 shared task on the identification of offensive language. In *Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018), Vienna, Austria – September 21, 2018. - Vienna, Austria*, pages 1–10, 2018.

48 Thilini Wijesiriwardene, Hale Inan, Ugur Kursuncu, Manas Gaur, Valerie L Shalin, Krishnaprasad Thirunarayan, Amit Sheth, and I Budak Arpinar. Alone: A dataset for toxic behavior among adolescents on twitter. In *International Conference on Social Informatics*, pages 427–439. Springer, 2020.

49 Ellery Wulczyn, Nithum Thain, and Lucas Dixon. Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th international conference on world wide web*, pages 1391–1399, 2017.

50 Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. Predicting the type and target of offensive posts in social media. *arXiv preprint*, 2019. `arXiv:1902.09666`.

51 Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2019, Minneapolis, MN, USA, June 6-7, 2019*, pages 75–86. Association for Computational Linguistics, 2019. `doi:10.18653/v1/s19-2010`.

52 Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020). In *Proceedings of SemEval*, 2020.

Bias in Knowledge Graphs – An Empirical Study with Movie Recommendation and Different Language Editions of DBpedia

Michael Matthias Voit
University of Mannheim, Germany

Heiko Paulheim
University of Mannheim, Germany

Abstract

Public knowledge graphs such as DBpedia and Wikidata have been recognized as interesting sources of background knowledge to build content-based recommender systems. They can be used to add information about the items to be recommended and links between those. While quite a few approaches for exploiting knowledge graphs have been proposed, most of them aim at optimizing the recommendation strategy while using a *fixed* knowledge graph. In this paper, we take a different approach, i.e., we fix the recommendation strategy and observe changes when using different underlying knowledge graphs. Particularly, we use different *language editions* of DBpedia. We show that the usage of different knowledge graphs does not only lead to differently biased recommender systems, but also to recommender systems that differ in performance for particular fields of recommendations.

2012 ACM Subject Classification Computing methodologies → Knowledge representation and reasoning; Information systems → Recommender systems

Keywords and phrases Knowledge Graph, DBpedia, Recommender Systems, Bias, Language Bias, RDF2vec

Digital Object Identifier 10.4230/OASIcs.LDK.2021.14

Related Version *Full Version*: `https://arxiv.org/abs/2105.00674`

Supplementary Material *Software (Source Code)*: `https://github.com/voitijaner/Movie-RSs-Master-Thesis-Submission-Voit`
archived at `swh:1:dir:5a1679a3579764cbd88c758be59c337e0f88a277`

1 Introduction

Large-scale knowledge graphs, such as DBpedia [22] and Wikidata [41], are recognized as a valuable ingredient for intelligent applications [16]. In such applications, they can provide background information on the entities processed, which often leads to performance improvements in downstream processing steps [37].

In the past, different works have been proposed on building recommender systems based on knowledge graphs, most prominently, DBpedia. The first of those approaches has probably been *dbrec*, dating back to 2010 [32]. Since then, a number of approaches have been proposed, challenges around the topic have been conducted [7], and recent approaches have been utilizing the omnipresent knowledge graph embeddings for computing recommendations [29, 39].

The vast majority of those works always utilizes a fixed knowledge graph (DBpedia in most cases) and then optimizes the recommendation algorithm to provide the best empirical results on a test dataset. This means that by fixing the knowledge graph upfront, the influence of the chosen graph, its coverage, data quality, and possible biases, are not examined.

In this paper, we postulate that the choice of a particular knowledge graph has an influence on the behavior of the overall system, and may lead to a certain bias. To analyze this bias, we train a recommendation system with a fixed setup and parameter settings based on the embedding method RDF2vec [38], using different versions of DBpedia, which have been extracted from Wikipedia language editions.

Assuming that the coverage, quality, and level of detail of recommended items (in our test scenario: movies) varies from language edition to language edition, we expect a certain bias to shine up when using different knowledge graphs. This is confirmed by our experiments, however, the bias is not as obvious as we expected. While the straight forward assumption is that, e.g., a recommender system based on the German DBpedia edition would develop a stronger bias towards recommending German films, the effects are more subtle than that, exposing different significant biases with respect to production country and genre.

The rest of this paper is structured as follows. Section 2 discusses related works. In Section 3, we lay out our experiment set up, followed by an analysis of findings in Section 4. We conclude with a summary and an outlook on future work.

2 Related Work

The two most well-known families of recommender systems are collaborative filtering and content-based recommender systems. While the former analyze the behavior of users and recommends items that are consumed by users with a similar behavior as the one at hand, the latter exploit similarities between the items per se. For that category of approaches, a model of the recommended items is required, which can be unstructured (e.g., a textual description) or structured (e.g., a set of attributes). [35]

For structured representations, public knowledge graphs like DBpedia or Wikidata have been recognized as a valuable source of information, since they already contain a large amount of information on various items in a structured form [16]. Most classic approaches use DBpedia and/or knowledge graphs tailored to the domain at hand, and base their decision on a similarity function based on a set of hand-picked attributes (e.g., genre and artist for music; genre, director, and actor for movies).

The first generation of recommender systems based on Knowledge Graphs, such as dbrec, were based on hand-picking attributes and relations. Later approaches also exploited automatic approaches for selecting attributes, either by adapting measures such as TF-IDF to graph data [8], or by using machine learning methods such as Random Forests, which can be used on larger feature sets and automatically identify the relevant ones [36].

The most recent generation of such recommender system utilizes *knowledge graph embeddings* [11]. Such embedding methods project resources in a knowledge graph into a lower-dimensional, numerical vector space. Since many of those projection methods lead to vector spaces in which similar resources are close to each other, distance in the embedding space can be exploited for recommendation, as depicted in figure 1. Such approaches, among others, have been analyzed for RDF2vec [39], metapath2vec [52], TransE [6, 19, 20, 44, 49], TransR [25, 40, 44, 48, 53], TransH [4, 44], TransD [14, 44], ComplEx [19], LINE [48], Laplacian Eigenmaps and node2vec [29, 31], and embedding methods specifically tailored to the recommendation task, like *RippleNet* [43], *CFKG* [54], *Hierarchical Collaborative Embedding* [56], *MKR* [46], and *UPM* [57]. More recently, graph neural networks have also gained a bit of traction [45, 47].

While we are aware that this set of examples for the usage of knowledge graphs for content-based recommender systems is far from complete, a common trend can be observed in almost all publications about such systems: they always fix the knowledge graph to be

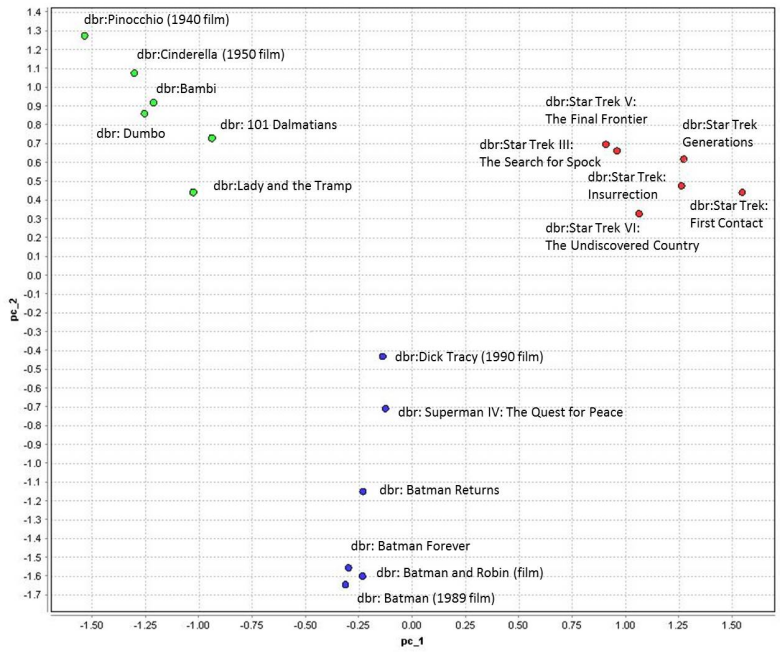


Figure 1 2-dimensional PCA projection of embedding vectors for a set of movies in DBpedia [39].

used upfront, and different variants are typically studied for the algorithms used to compute similarities, but not for the graph as such. In the rare cases where results obtained with multiple knowledge graphs are examined (e.g., [39], which contrasts results based on DBpedia and Wikidata), they are only compared based on the scoring function (e.g., F1 score), but other influences on the behavior of the recommender system are not analyzed. Hence, the influence of the choice for a particular knowledge graph is still underexplored.

In this paper, we conduct a study to shed some light on that aspect. To that end, we use different versions of DBpedia. DBpedia is extracted from Wikipedia infoboxes by the use of mappings to a central ontology. There are versions of DBpedia for different Wikipedia languages, called *DBpedia language editions* [22].

It is known that language editions of Wikipedia differ in coverage and level of detail. Their size ranges from a few hundred to a few million articles.[1] These differences have been analyzed with respect to various aspects, e.g., topical coverage [1, 9, 15, 28], article quality [24] and neutrality [3, 26, 51, 55], bias related to geography [2] or gender [42], and user behavior [12, 23].

The difference in the quality of infobox data in different Wikipedia language editions has also been studied [50]. This is particularly interesting for our scenario, since the DBpedia knowledge graph draws its information from those infoboxes. Hence, in the light of those studies, we expect significant differences in knowledge graphs extracted from different Wikipedia languages, and we want to explore in how far they lead to difference in downstream applications such as recommender systems.

[1] `https://en.wikipedia.org/wiki/List_of_Wikipedias`

Table 1 Statistic of common movies in different language version with the movies mapped to the English DBpedia.

	it	pl	es	pt	fr	de	ru	nl	ja
# Movies	**24k**	13k	12k	12k	**16k**	**19k**	**15k**	10k	10k
Intersection with MovieLens 1M mapped to DBpedia-en	**2,610**	2.106	2,019	2,092	**2,658**	**2,426**	**2,255**	1,793	1,888

3 Experimental Setup

In our experiments, we use the MovieLens 1M dataset [13], which contains a 1M 1-5 star ratings by 6,040 users for 3,952 movies. Moreover, we use DBpedia version 2016-10.[2]

In earlier works, links from MovieLens 1M to DBpedia have been provided [30]. Since DBpedia has been evolving since the original linking was performed, we removed all links that refer either to entities which do not exist anymore (i.e., the URI has changed in later releases of DBpedia) or to entities derived from disambiguation pages. Our resulting dataset consists of 3,123 movies linked to the English DBpedia.

3.1 Datasets

To investigate the influence of the usage of different versions of DBpedia on a recommender system, we utilize different language versions of DBpedia. In a preliminary study, we looked at the ten largest language editions of DBpedia[3], and analyzed the overlap with the 3,123 movies in our dataset linked to the English DBpedia. To that end, we utilze the links between DBpedia versions, which are extracted from the inter-language links in Wikipedia. The results are depicted in Table 1.

To ensure a reasonable coverage, we decided to use the five dataset which have the most information about movies and the highest overlap with the English dataset. Hence, we decided to base our analysis on English, Italian, French, German, and Russian. The subset of the original 3,123 movies which have a corresponding entity in all five datasets comprises 1,948 movies.

We apply two additional filtering steps, as suggested by [5], [30], and [38]. To avoid a popularity bias, the top-rated 1% of all movies are removed. In the second step, users with less than 50 ratings are removed, and so are movies without any ratings. After this step, we obtain a dataset with 1,918 movies, 675,960 ratings, and 3,642 users. This set is used as the basis for all our experiments.

3.2 Recommender Algorithm

As discussed above, in our experiments, we aim at keeping the recommender algorithm fixed, while varying the underlying knowledge graph. We intentionally use a simple algorithm for the recommendations, as our goal is not to maximize the performance of the recommendation as such, but to examine the influence of the underlying knowledge graph.

[2] https://wiki.dbpedia.org/develop/datasets/dbpedia-version-2016-10
[3] https://wiki.dbpedia.org/services-resources/datasets/dataset-statistics

Following the setup in [38], we use RDF2vec to compute vector space embeddings of the different DBpedia graphs. RDF2vec extracts random walks from a knowledge graph, which are represented as "sentences" of entities and predicates in the knowledge graph. On that set of sentences, the word2vec algorithm [27] is run, which then computes an embedding vector for each entity (and predicate).

We computed RDF2vec embeddings for the five DBpedia language editions identified above, using the best performing parameter setting identified in [38], i.e., extracting 500 walks per entity with a depth of 4, a dimensionality of 200 for the word2vec model, using the Skip-Gram variant.[4]

The similarity of two movies is then computed as the cosine similarity of the corresponding vectors in that vector space. To that end, we compute a score y_{uj} for an unrated movie j and user u is calculated with the following formula:

$$y_{uj} = \frac{\sum_{i \in I_u} cos(i,j) * r_{ui}}{\sum_{i \in I_u} cos(i,j)} \tag{1}$$

Here, I_u are the previous observations from user u and r_{ui} denotes the rating of item i of the user u in the training set. Then, the N movies with the highest scores are returned for each user. For this procedure, we used the item similarity recommender of the GraphLab Create python framework[5].

3.3 Metrics

To evaluate the quality of recommendations, we use the standard measures of recall, precision, and F1 score. Here, recall measures the fraction of items that a user ultimately rated positively which were recommended to him or her, and precision measures the fraction of recommended items which were ultimately rated positively. F1 is the harmonic mean between the two.

Besides the quality of recommendations, we are interested in differences among recommendations created based on different knowledge graphs. To that end, we look at different *categorical features*, like language or genre. For a categorical feature C (such as *language*), we can compute the probability of recommendations with a certain feature value c (such as *German*), i.e.,

$$p(c) = \frac{|R_c|}{|R|} \tag{2}$$

where $|R_c|$ is the total number of items that were recommended by a certain approach which have the categorical feature c, and $|R|$ is the total number of recommendations computed. These probabilities can then be compared for recommender systems based on different knowledge graphs.

In order to distinguish random variations from effects actually induced by the use of different knowledge graphs, we additionally conduct a chi-squared test:

$$\chi^2_{KG} = \sum_{c \in C} \frac{(|R_c| - (c_e * |R|))^2}{(c_e * |R|)} \tag{3}$$

[4] All code and data is available online at `https://github.com/voitijaner/Movie-RSs-Master-Thesis-Submission-Voit`

[5] `https://turi.com/products/create/docs/generated/graphlab.recommender.item_similarity_recommender.ItemSimilarityRecommender.html`

Table 2 Performance of the recommender systems per KG.

KG	Precision	Recall	F1 score
de	0.057	0,0404	0.047
fr	0.054	0.038	0.044
en	0.053	0.038	0.044
it	0.048	0.036	0.042
ru	0.042	0.028	0.034

where c_e is the expected value of recommendations with a categorical feature value c. We sum up the χ^2_{KG} values for all KGs, and compare them against the χ^2 distribution with $(|KGs| - 1) \cdot (|C| - 1)$ degrees of freedom, and an α value of 0.05 to test for significance. All the results presented below are not a random result according to that definition.

4 Findings and Observations

In total, we compare five recommender systems, based on the five different knowledge graphs. We analyze both the overall performance, as well as biases w.r.t. production countries and genres.

4.1 Overall Performance

In a first analysis, we look at the overall performance difference between the recommenders based on the five knowledge graphs. We can see that the one based on the German DBpedia works best, which is most likely due to a higher linkage degree for movies.

Most strikingly, the English DBpedia – which is used as the basis for the majority of works which claim to use "DBpedia" as a source of background knowledge – performs worse than its German and French counterpart. This shows that this choice, which is often done by simple heuristics such as popularity and availability, might not be an optimal one.

4.2 Bias for Production Countries

The first analysis for bias we perform is whether certain recommenders have stronger tendencies to recommend movies with a particular production country. The underlying hypothesis is that recommenders based on a knowledge graph derived from a Wikipedia in a particular language will have a tendency to also recommend more movies from a production country where that language is spoken (e.g., the recommender based on German DBpedia could have a stronger tendency to recommend German or Austrian movies).

Table 3 shows the top 10 production countries in the dataset. It can be observed that the dataset is heavily skewed towards movies from the USA and, to the lesser extent, UK, whereas other production countries only play a minor role.

Table 4 shows the fraction of movies from the top 10 production countries recommended by the systems based on the different knowledge graphs. We can see that the massive skew of the dataset towards US movies is also reflected in the results: except for the US movies, the majority of recommendations is below the expected value c_e.

Furthermore, it can be observed that, although significant differences in the behavior exist, there is no clear pattern of the form that follows the above mentioned hypothesis. Except for movies from the US and Australia, the peak of recommendations is always observed for a KG which is not in the respective language. For example, the fraction of German

Table 3 Top 10 production countries in the dataset.

Country	# of movies	# of ratings
USA	1,679	622,946
UK	267	92,470
France	127	27,362
Germany	69	21,170
Italy	62	10,887
Canada	46	12,367
Australia	30	13,330
Japan	26	5,718
Spain	16	3,813
Mexico	15	6,790

Table 4 Fraction of recommendations for different production countries by knowledge graph. c_e denotes the expected fraction based on the prevalence in the dataset.

Country/KG	de	fr	it	ru	en	c_e
USA	0.728	0.750	0.762	0.761	**0.782**	0.744
UK	**0.136**	0.143	0.098	0.091	0.108	0.110
France	0.028	0.030	0.036	**0.037**	0.026	0.033
Germany	0.012	0.018	0.012	0.030	**0.034**	0.025
Italy	**0.016**	0.009	0.013	0.009	0.009	0.013
Canada	0.020	0.009	**0.021**	0.005	0.006	0.015
Australia	0.017	0.010	0.013	0.008	**0.020**	0.016
Japan	0.006	0.005	**0.012**	0.004	0.006	0.007
Spain	0.006	0.004	**0.006**	0.002	0.005	0.005
Mexico	0.004	0.001	0.005	**0.006**	0.002	0.008

movies recommended by the system based on the English DBpedia is almost three times as high as the one based on the German DBpedia. Also, for other languages, the patterns are different: the highest fraction of French movies is recommended by the system based on the Russian KG, the highest fraction of Italian movies is recommended by the system based on the German KG, and so on.

4.3 Bias for Genres

In a second analysis, we inspect another possible bias induced by the different KGs, i.e., the bias to recommend movies from particular genres. Table 5 shows the top 10 genres in the dataset.

Table 6 shows the recommendations based on the different knowledge graphs for the top 10 genres. Here, we can again observe some interesting deviations. The recommender based on the Russian DBpedia has a tendency towards action, science fiction, and adventure movies, while the one based on the Italian DBpedia tends to recommend more movies from the comedy, thriller, and romance genres. Those findings partially correlate with studies on the popularity of particular genres in different countries. In [10], the authors discuss that, e.g., action movies are more popular in Russia than in English-speaking or European countries, and that comedy movies are more popular in Italy. Hence, it is likely that a local Wikipedia community in those countries will put more emphasis on editing movies in

Table 5 Top 10 genres in the dataset.

Genre	# of movies	# of ratings
Drama	721	174,635
Comedy	562	184,700
Action	351	133,342
Thriller	320	74,457
Romance	244	44,784
Horror	225	22,700
Science Fiction	183	52,648
Adventure	172	36,827
Children's	130	10,316
Crime	115	7,621

Table 6 Fraction of recommendations for different genres by knowledge graph. c_e denotes the expected fraction based on the prevalence in the dataset.

Genre/KG	de	fr	it	ru	en	c_e
Drama	**0.198**	0.170	0.187	0.172	0.190	0.162
Comedy	0.191	0.192	**0.207**	0.198	0.166	0.168
Action	0.089	0.010	0.074	**0.129**	0.112	0.123
Thriller	0.072	0.086	**0.097**	0.088	0.084	0.095
Romance	0.073	0.055	**0.081**	0.080	0.052	0.071
Horror	0.043	0.050	0.044	0.043	**0.053**	0.043
Science Fiction	0.055	0.045	0.044	**0.056**	0.053	0.073
Adventure	0.053	0.045	0.053	**0.070**	0.049	0.063
Children's	0.041	**0.053**	0.052	0.026	0.046	0.031
Crime	0.029	0.039	0.025	0.044	**0.045**	0.038

the respective genre in Wikipedia, which then leads those movies being more and better represented in the corresponding language-specific DBpedia, and ultimately a stronger bias of the recommender system based on that knowledge graph towards that genre.

4.4 Specific Performance Differences

The observation that recommenders based on different knowledge graphs expose biases towards particular genres also leads us to looking at the problem from a different angle. In particular, we want to analyze if recommenders based on different knowledge graphs work better or worse for single genres. To that end, we created partitions of our dataset by movie genre, and ran the recommender systems on those partitions. Overall, runs for ten different genres were performed.

The results are depicted in Table 7. We can observe that there are rather strong differences between the genre-specific recommender performance. The French DBpedia, which has also been identified as the best source of background knowledge above, yields superior results for half of the genres. On the other hand, also the Russian DBpedia, which shows the worst overall performance, outperforms all other recommender systems on the crime genre.

The differences on the individual genres are sometimes marginal, but for some genres (e.g., horror, children's), the best performing system can achieve results which are twice or even thrice as high as those which perform worst. This shows that there is no one-size-fits-all solution, and that the exploration of different knowledge graphs for a particular task and domain is at least as beneficial as the exploration of algorithmic alternatives.

Table 7 Performance (F1) of recommenders for different genres by knowledge graph.

Genre/KG	de	fr	it	ru	en
Drama	0.040	**0.045**	0.034	0.030	0.040
Comedy	**0.078**	0.067	0.055	0.053	0.068
Action	0.091	**0.114**	0.089	0.080	0.105
Thriller	0.083	**0.085**	0.061	0.064	0.080
Romance	0.038	0.046	0.036	0.043	**0.056**
Horror	0.073	0.072	0.066	0.040	**0.082**
Science Fiction	0.101	**0.124**	0.106	0.080	0.095
Adventure	0.090	**0.115**	0.093	0.097	0.082
Children's	**0.209**	0.146	0.176	0.064	0.200
Crime	0.097	0.098	0.084	**0.121**	0.099

5 Conclusion and Future Work

In this paper, we have conducted a comparative study of recommender systems based on different knowledge graphs, particularly: versions of DBpedia, based on Wikipedia in different languages. The experiment design has been chosen in a way that a basic recommendation strategy was chosen and fixed, and five different underlying knowledge graphs were used. The results show that there are considerable differences in preferences of the recommenders. Particularly, we analyzed production countries and genres, but our method is generally applicable to other categorical variables as well (e.g., gender of producer or director, low, medium and high budget, etc.).

The second major observation is that despite overall trends, not all knowledge graphs are equally well suited for particular recommendation tasks. When building a recommender system for movies from a particular genre, the globally best performing knowledge graph might not be the one which performs best locally on a given task. Here, we argue that the choice of a knowledge graph – which is usually fixed upfront in most related works – should be treated equally, if not even more important as fine tuning algorithms.

The problem of fixing a knowledge graph upfront is not limited to recommender systems. Knowledge graphs have been suggested to be used in other fields as well, such as explainable AI [21], data interpretation [33], or social media analysis [34]. Like for recommender systems, biases induced by the choice of a particular knowledge graph have not been researched to a large extent here.

In the future, we see a few interesting directions to pursue. One of those is the extension of the analysis both to other domains, such as music or book recommendations, as well as the inclusion of further categorical variables, such as biases towards male or female authors, or black or white musicians.

The inclusion of further knowledge graphs in studies like these is also an interesting area. With the advent of more cross-domain knowledge graphs, such as Wikidata [41], CaLiGraph [17], and DBkWik [18] we assume that each of those comes with its very own coverage biases, and a setup like the one discussed in this paper would be a way of systemically investigating the possible impact of such biases on downstream applications. Furthermore, it is an open question whether combining information from different knowledge graphs is a suitable way of reducing the individual biases.

Finally, while we argue that the selection of a particular knowledge graph is at least as important as the selection and fine-tuning of a recommender algorithm, interaction effects between the two decisions must not be neglected. We assume that, while there is no one-size-

fits-all solution either on the knowledge graph nor on the algorithm side, the sweet spot for an optimal solution might not just be the straight forward combination of the knowledge graph and algorithm which perform best in isolation.

References

1 Patti Bao, Brent Hecht, Samuel Carton, Mahmood Quaderi, Michael Horn, and Darren Gergle. Omnipedia: bridging the wikipedia language gap. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1075–1084, 2012.

2 Pablo Beytía. The positioning matters: Estimating geographical bias in the multilingual record of biographies on wikipedia. In *Companion Proceedings of the Web Conference 2020*, pages 806–810, 2020.

3 Ewa S Callahan and Susan C Herring. Cultural bias in wikipedia content on famous persons. *Journal of the American society for information science and technology*, 62(10):1899–1915, 2011.

4 Yixin Cao, Xiang Wang, Xiangnan He, Zikun Hu, and Tat-Seng Chua. Unifying knowledge graph learning and recommendation: Towards a better understanding of user preferences. In *The world wide web conference*, pages 151–161, 2019.

5 Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 39–46, 2010.

6 Amine Dadoun, Raphaël Troncy, Olivier Ratier, and Riccardo Petitti. Location embeddings for next trip recommendation. In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 896–903, 2019.

7 Tommaso Di Noia, Iván Cantador, and Vito Claudio Ostuni. Linked open data-enabled recommender systems: Eswc 2014 challenge on book recommendation. In *Semantic Web Evaluation Challenge*, pages 129–143. Springer, 2014.

8 Tommaso Di Noia, Roberto Mirizzi, Vito Claudio Ostuni, and Davide Romito. Exploiting the web of data in model-based recommender systems. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 253–256, 2012.

9 Young-Ho Eom, Pablo Aragón, David Laniado, Andreas Kaltenbrunner, Sebastiano Vigna, and Dima L Shepelyansky. Interactions of cultures and top people of wikipedia from ranking of 24 language editions. *PloS one*, 10(3):e0114825, 2015.

10 Stephen Follows. The relative popularity of genres around the world:, 2016. URL: `https://stephenfollows.com/relative-popularity-of-genres-around-the-world`.

11 Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, 2020.

12 Noriko Hara, Pnina Shachaf, and Khe Foon Hew. Cross-cultural analysis of the wikipedia community. *Journal of the American Society for Information Science and Technology*, 61(10):2097–2108, 2010.

13 F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015.

14 Ming He, Bo Wang, and Xiangkun Du. Hi2rec: Exploring knowledge in heterogeneous information for movie recommendation. *IEEE Access*, 7:30276–30284, 2019.

15 Brent Hecht and Darren Gergle. Measuring self-focus bias in community-maintained knowledge repositories. In *Proceedings of the fourth international conference on communities and technologies*, pages 11–20, 2009.

16 Nicolas Heist, Sven Hertling, Daniel Ringler, and Heiko Paulheim. *Knowledge Graphs on the Web – an Overview*, pages 3–22. IOS Press, 2020.

17 Nicolas Heist and Heiko Paulheim. Uncovering the semantics of wikipedia categories. In *International semantic web conference*, pages 219–236. Springer, 2019.

18 Sven Hertling and Heiko Paulheim. Dbkwik: A consolidated knowledge graph from thousands of wikis. In *2018 IEEE International Conference on Big Knowledge (ICBK)*, pages 17–24. IEEE, 2018.

19 Hen-Hsen Huang. An mpd player with expert knowledge-basedsingle user music recommendation. In *IEEE/WIC/ACM International Conference on Web Intelligence-Companion Volume*, pages 318–321, 2019.

20 Jin Huang, Wayne Xin Zhao, Hongjian Dou, Ji-Rong Wen, and Edward Y Chang. Improving sequential recommendation with knowledge-enhanced memory networks. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 505–514, 2018.

21 Freddy Lecue. On the role of knowledge graphs in explainable ai. *Semantic Web*, 11(1):41–51, 2020.

22 Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic web*, 6(2):167–195, 2015.

23 Florian Lemmerich, Diego Sáez-Trumper, Robert West, and Leila Zia. Why the world reads wikipedia: Beyond english speakers. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pages 618–626, 2019.

24 Włodzimierz Lewoniewski, Krzysztof Węcel, and Witold Abramowicz. Quality and importance of wikipedia articles in different languages. In *International Conference on Information and Software Technologies*, pages 613–624. Springer, 2016.

25 Qika Lin, Yaoqiang Niu, Yifan Zhu, Hao Lu, Keith Zvikomborero Mushonga, and Zhendong Niu. Heterogeneous knowledge-based attentive neural networks for short-term music recommendations. *IEEE Access*, 6:58990–59000, 2018.

26 Paolo Massa and Federico Scrinzi. Manypedia: Comparing language points of view of wikipedia communities. In *Proceedings of the Eighth Annual International Symposium on Wikis and Open Collaboration*, pages 1–9, 2012.

27 Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

28 Volodymyr Miz, Joëlle Hanna, Nicolas Aspert, Benjamin Ricaud, and Pierre Vandergheynst. What is trending on wikipedia? capturing trends and language biases across wikipedia editions. In *Companion Proceedings of the Web Conference 2020*, pages 794–801, 2020.

29 Cataldo Musto, Pierpaolo Basile, and Giovanni Semeraro. Embedding knowledge graphs for semantics-aware recommendations based on dbpedia. In *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, pages 27–31, 2019.

30 Tommaso Di Noia, Vito Claudio Ostuni, Paolo Tomeo, and Eugenio Di Sciascio. Sprank: Semantic path-based ranking for top-n recommendations using linked open data. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(1):1–34, 2016.

31 Enrico Palumbo, Giuseppe Rizzo, and Raphaël Troncy. Entity2rec: Learning user-item relatedness from knowledge graphs for top-n item recommendation. In *Proceedings of the eleventh ACM conference on recommender systems*, pages 32–36, 2017.

32 Alexandre Passant. dbrec – music recommendations using dbpedia. In *International Semantic Web Conference*, pages 209–224. Springer, 2010.

33 Heiko Paulheim. Generating possible interpretations for statistics from linked open data. In *Extended Semantic Web Conference*, pages 560–574. Springer, 2012.

34 Guangyuan Piao and John G Breslin. Exploring dynamics and semantics of user interests for user modeling on twitter for link recommendations. In *proceedings of the 12th international conference on semantic systems*, pages 81–88, 2016.

35 Francesco Ricci, Lior Rokach, and Bracha Shapira. Recommender systems: introduction and challenges. In *Recommender systems handbook*, pages 1–34. Springer, 2015.

36 Petar Ristoski, Eneldo Loza Mencía, and Heiko Paulheim. A hybrid multi-strategy recommender system using linked open data. In *Semantic Web Evaluation Challenge*, pages 150–156. Springer, 2014.

37 Petar Ristoski and Heiko Paulheim. Semantic web in data mining and knowledge discovery: A comprehensive survey. *Journal of Web Semantics*, 36:1–22, 2016.

38 Petar Ristoski, Jessica Rosati, Tommaso Di Noia, Renato De Leone, and Heiko Paulheim. Rdf2vec: Rdf graph embeddings and their applications. *Semantic Web*, 10(4):721–752, 2019.

39 Jessica Rosati, Petar Ristoski, Tommaso Di Noia, Renato de Leone, and Heiko Paulheim. Rdf graph embeddings for content-based recommender systems. In *CEUR workshop proceedings*, volume 1673, pages 23–30. RWTH, 2016.

40 Xiaoli Tang, Tengyun Wang, Haizhi Yang, and Hengjie Song. Akupm: Attention-enhanced knowledge-aware user preference model for recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1891–1899, 2019.

41 Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014.

42 Claudia Wagner, David Garcia, Mohsen Jadidi, and Markus Strohmaier. It's a man's wikipedia? assessing gender inequality in an online encyclopedia. *arXiv preprint*, 2015. `arXiv:1501.06307`.

43 Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 417–426, 2018.

44 Hongwei Wang, Fuzheng Zhang, Xing Xie, and Minyi Guo. Dkn: Deep knowledge-aware network for news recommendation. In *Proceedings of the 2018 world wide web conference*, pages 1835–1844, 2018.

45 Hongwei Wang, Fuzheng Zhang, Mengdi Zhang, Jure Leskovec, Miao Zhao, Wenjie Li, and Zhongyuan Wang. Knowledge-aware graph neural networks with label smoothness regularization for recommender systems. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 968–977, 2019.

46 Hongwei Wang, Fuzheng Zhang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. Multi-task feature learning for knowledge graph enhanced recommendation. In *The World Wide Web Conference*, pages 2000–2010, 2019.

47 Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. Knowledge graph convolutional networks for recommender systems. In *The world wide web conference*, pages 3307–3313, 2019.

48 Meng Wang, Mengyue Liu, Jun Liu, Sen Wang, Guodong Long, and Buyue Qian. Safe medicine recommendation via medical knowledge graph embedding. *arXiv preprint*, 2017. `arXiv:1710.05980`.

49 Xinyu Wang, Ying Zhang, Xiaoling Wang, and Jin Chen. A knowledge graph enhanced topic modeling approach for herb recommendation. In *International Conference on Database Systems for Advanced Applications*, pages 709–724. Springer, 2019.

50 Krzysztof Węcel and Włodzimierz Lewoniewski. Modelling the quality of attributes in wikipedia infoboxes. In *International Conference on Business Information Systems*, pages 308–320. Springer, 2015.

51 Hartmut Wessler, Christoph Kilian Theil, Heiner Stuckenschmidt, Angelika Storrer, and Marc Debus. Wikiganda: Detecting bias in multimodal wikipedia entries. *New Studies in Multimodality. London/New York: Bloomsbury*, pages 201–224, 2017.

52 Deqing Yang, Zikai Guo, Ziyi Wang, Juyang Jiang, Yanghua Xiao, and Wei Wang. A knowledge-enhanced deep recommendation framework incorporating gan-based models. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 1368–1373. IEEE, 2018.

53 Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 353–362, 2016.

54 Yongfeng Zhang, Qingyao Ai, Xu Chen, and Pengfei Wang. Learning over knowledge-base embeddings for recommendation. *Algorithms*, 11(9), 2018.

55 Yiwei Zhou, Elena Demidova, and Alexandra I Cristea. Who likes me more? analysing entity-centric language-specific bias in multilingual wikipedia. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, pages 750–757, 2016.

56 Zili Zhou, Shaowu Liu, Guandong Xu, Xing Xie, Jun Yin, Yidong Li, and Wu Zhang. Knowledge-based recommendation with hierarchical collaborative embedding. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 222–234. Springer, 2018.

57 Guiming Zhu, Chenzhong Bin, Tianlong Gu, Liang Chang, Yanpeng Sun, Wei Chen, and Zhonghao Jia. A neural user preference modeling framework for recommendation based on knowledge graph. In *Pacific Rim International Conference on Artificial Intelligence*, pages 176–189. Springer, 2019.

Enriching Word Embeddings with Food Knowledge for Ingredient Retrieval

Álvaro Mendes Samagaio
Faculty of Engineering, University of Porto, Portugal
Fraunhofer Portugal, Porto, Portugal

Henrique Lopes Cardoso
Faculty of Engineering, University of Porto, Portugal
Artificial Intelligence and Computer Science Laboratory (LIACC), Porto, Portugal

David Ribeiro
Fraunhofer Portugal, Porto, Portugal

Abstract

Smart assistants and recommender systems must deal with lots of information coming from different sources and having different formats. This is more frequent in text data, which presents increased variability and complexity, and is rather common for conversational assistants or chatbots. Moreover, this issue is very evident in the food and nutrition lexicon, where the semantics present increased variability, namely due to hypernyms and hyponyms. This work describes the creation of a set of word embeddings based on the incorporation of information from a food thesaurus – *LanguaL* – through retrofitting. The ingredients were classified according to three different facet label groups. Retrofitted embeddings seem to properly encode food-specific knowledge, as shown by an increase on accuracy as compared to generic embeddings (+23%, +10% and +31% per group). Moreover, a weighing mechanism based on TF-IDF was applied to embedding creation before retrofitting, also bringing an increase on accuracy (+5%, +9% and +5% per group). Finally, the approach has been tested with human users in an ingredient retrieval exercise, showing very positive evaluation (77.3% of the volunteer testers preferred this method over a string-based matching algorithm).

2012 ACM Subject Classification Computing methodologies → Artificial intelligence; Computing methodologies → Knowledge representation and reasoning; Computing methodologies → Lexical semantics

Keywords and phrases Word embeddings, Retrofitting, LanguaL, Food Embeddings, Knowledge Graph

Digital Object Identifier 10.4230/OASIcs.LDK.2021.15

Funding *Henrique Lopes Cardoso*: This research is partially supported by LIACC (FCT/UID/CEC/0027/2020), funded by Fundação para a Ciência e a Tecnologia (FCT).

1 Introduction

Conversational agents and smart assistants are an interesting opportunity for many application areas [32]. Fostered by the latest advances in artificial intelligence and natural language processing [12], these software allow interaction with computer systems through conversation or chat interfaces [6]. From an interaction point of view, they enable intuitive interaction to access different services. Conversational agents are also cost-effective and may, in may cases, replace human labour [13] in providing access to services from simple access to information to more complex services including infotainment, customer support [34], and recommendation systems. Smart assistants may also positively impact user health by interfacing with health-related services [4, 1].

In the context of food and nutrition, conversational agents may interface with systems that help its users acquiring healthier eating habits, inform and support their decisions [8] which may help preventing several chronic diseases [9]. One of the challenges in developing

conversational agents for this domain is related to the complexity of the food taxonomy, which is rich in synonyms, hypernyms and hyponyms [16, 30]. For example, when users refer to *cheese*, they are mentioning a large group of different types of cheese and any of them could be considered. This work could be very useful for tasks where the system must match different sources of information. In this case the goal was to match a query in the form of a named entity with all the entries in a database.

In this work, we describe an approach that exploits semantic knowledge from the food domain in a food matching algorithm. More specifically, we explore the enrichment of pre-trained word embeddings with food-domain-specific knowledge, which can then be used in a number of tasks. To the best of our knowledge, this is the first set of word vectors that truly incorporate semantic information from a knowledge graph focused in food-related concepts.

2 Related Work

This section explores the work that has been done and is currently applied regarding word representations and semantic knowledge.

There are several works that study word representations in high dimensional spaces, mainly focusing on capturing context from large corpora. The underlying premise is that contextual information constitutes a proper representation of linguistic items. Word representations have gained increased attention in NLP tasks with the work of Mikolov et al. [17] (Word2Vec). Other sets of pre-trained embeddings use slightly different techniques to capture and encode the semantic and contextual information in texts. *GloVe* [19] is trained on global word-word co-ocurrence statistics aggregated from a corpus. It uses a log-bilinear regression model to create the word vectors, hence combining the features that come from global matrix factorization and also from local context windows.

It is possible to find pre-trained lexicons created through the application of Word2Vec or GloVe-like algorithms to huge corpora (such as the Google News Dataset). These lexicons are generic enough to capture the meanings of the words. However, this can be seen as one of their greatest disadvantages, which is the fact that they were not trained specifically for a given domain, and thus may not represent well enough the semantics of that domain. More recently, the development of Language Models that make use of transformer architectures [31], such as BERT [7] or ELMo [20], shifted the state-of-the-art on word representations for NLP, from the pre-trained and static embeddings to contextualized embeddings. These models use self-attention layers that change the embeddings of each word according to its context. In this case, the same word in different contexts will be represented by different vectors.

Besides word vector representations, there are other resources that can be used to portray concepts and their relations. Examples include knowledge graphs or ontologies that encode semantic relations in a graph which connects concepts that are linked in some way. There is a wide range of knowledge graphs available and they can also be used by conversational agents to retrieve information according to a given query [5, 3, 4]. It is known that commercially available smart assistants such as *Google Assistant* and *Siri* use knowledge graphs to process the information inputted by the user in order to retrieve the correct answers [15]. One of the most well-known knowledge graphs is *WordNet* [18], a lexical database for the English language, that also includes synonyms and definitions for words. It is widely used to improve the performance on NLP tasks and applications. Another general-purpose knowledge graph that is publicly available is the *ConceptNet* [28], created as part of the Open Mind Common Sense project [11]. This knowledge graph has multilingual properties, where the

same concept in two different languages share a common semantic space, which is informed by all other languages. Knowledge graphs have the disadvantage of not being as easy to use as word embeddings; however, it is possible to incorporate this information into word embeddings [26, 35, 27], in order to improve the semantic relations between connected concepts. Speer et al. [28] make use of a set of pre-trained word embeddings, *ConceptNet Numberbatch*, that have been fine-tuned to encompass the relations present in *ConceptNet*, benefiting from the fact that they include semi-structured common sense knowledge. It was built on a combination of data from *ConceptNet*, *Word2Vec* vectors, *GloVe* vectors and *OpenSubtitles*, through a technique called *retrofitting*, to inject the knowledge into the vectors. Another interesting aspect of *ConceptNet Numberbatch* is that the multilingual properties from *ConceptNet* are kept in the embeddings, making it a very interesting resource for multilingual applications.

Retrofitting is a technique firstly introduced by Faruqui el al. [10] and, as previously mentioned, aims at incorporating the data present in semantic lexicons such as *WordNet* or *ConceptNet* into a previously defined word vector space, such as *Word2Vec* or *GloVe*. Hence, this refines the vector space to account for relational information, meaning that words which are lexically linked together should have similar vector representations. Retrofitting works by applying a linear vector transformation to the vectors that closes the gap between related words and increases the distance between lexically unrelated words. The transformation leads to a loss function Ψ that should be minimized, represented in Equation 1, where $\hat{q}_i$ is the initial vector, q_i the retrofitted vector and q_j its neighbors in the ontology. The parameters α and β are hyperparameters that control the relative strength of each parcel in the equation.

$$\Psi = \sum_{i=1}^{n} \left[\alpha_i \left\| q_i - \hat{q}_i \right\|^2 + \sum_{(i,j)\in E} \beta_{ij} \left\| q_i - q_j \right\|^2 \right] \tag{1}$$

In order to minimize the loss represented in Equation 1, it must be differentiated, resulting in Equation 2, which corresponds to the linear transformation applied to the vectors.

$$q_i = \frac{\sum_{j:(i,j)\in E} \beta_{ij} q_j + \alpha_i \hat{q}_i}{\sum_{j:(i,j)\in E} \beta_{ij} + \alpha_i} \tag{2}$$

By carefully analyzing the equation, one can conclude that it corresponds to the weighted average of the initial vector embedding and the vectors representing the concepts that are linked to it, controlled by parameters α and β, where the former attributes increased relevance to the initial vector and the latter controls the importance of the linked concepts.

Regarding food specific embeddings, Food2Vec[1] is a set of pre-trained embeddings that were generated using a corpus of recipe instructions. The goal was to create a recipe recommendation system that joined ingredients in order to create new recipes based on the embeddings of those ingredients. Although the embeddings are specialized for food, they do not capture the semantic relationships between hypernyms and hyponyms among ingredients. Instead, these embeddings encoded the relations that several ingredients have when used together. As an example, according to this methodology, *parmesan* is closer to *pasta* than to *cheddar*, even though *parmesan* and *cheddar* are two types of cheese. This is due to the fact that *parmesan* and *pasta* are used together many times in recipes, whereas it is rather rare to mix *parmesan* and *cheddar* in the same recipe. This approach would not be appropriate

[1] `https://jaan.io/food2vec-augmented-cooking-machine-intelligence/`

for ingredient classification, although it is an interesting approach to new recipes. A similar approach is followed by Tansey et al. [29], which encodes complete diets into a vector space using a combination of Word2Vec embeddings and nutritional information. Moreover, the work of Sauer et al. [24] tries to emulate different ingredient flavors in a vector space. Also, the work of Popovski [21] tries to generate a set of embeddings for a food ontology; however, the authors were not able to really grasp and capture the semantic relations between the concepts, since they only used information from the knowledge graph itself. That being said, there is a lack of NLP tools for food-related applications that are actually able to incorporate and represent the semantic relations between the different concepts and items.

3 Methodology

This work is part of the implementation of a conversational agent into a nutritional recommender system [22, 23], part of the *LiFANA* project [2]: a smart meal planner that takes into account personal information and preferences to create meal plans that are tailored to the user's nutritional needs. The recommendation engine relies on a recipe database that was created by nutritionists based on the McCance and Widdowson's The Composition of Foods[2] integrated dataset [25]. This database contains the nutritional composition of different food and its corresponding classification using *LanguaL* [14][3] descriptors. *LanguaL* is a multilingual thesaurus that allows describing food from different facts. Each food is described by a set of descriptors from different facets pertaining to different perspectives to classify food, including for instance its classification, source and presentation method. A numeric coding is used which allows translating each concept in different languages [14]. The recipes used by the system were built on top of this information by mixing ingredients and quantities. Every ingredient present in the database has at least one *LanguaL* descriptor as a classifier. In this work only facets A (food groups), B (origin) and C (part of animal or plant where the ingredient comes from) are considered, since they are regarded as the ones that effectively describe the ingredient when considering food preferences:

- Facet A – The ingredients are classified according to a food group to which they belong. Facet A gathers several international standards for food grouping. For the purpose of this work, the classification from the European Food Groups will be considered, since it was regarded as the one with the most granularity.
- Facet B – Addresses the food source and has several hierarchical levels. For this work, only the last and more specific level is considered. As examples, *milk*'s food source is *cow* and *raisin*'s food source is *grape*. This facet is particularly important to aggregate foods that correspond to their food source, such as fruits and vegetables or types of fish.
- Facet C – Categorizes the part of the animal or plant from which the ingredient is extracted. To illustrate, the descriptor for *cheese* under this classification is *milk*. This facet presents the least connection to current language terms, although it can be important to discern from similar ingredients semantically.

An example ingredient classification is illustrated below:

- *Chicken curry, chilled/frozen, reheated*
  LanguaL Descriptors:
  - **A0715** 25 Poultry and Poultry Products (EFG)
  - **B1457** Chicken
  - **C0268** Skeletal Meat Part, Without Bone, Without Skin

[2] https://www.gov.uk/government/publications/composition-of-foods-integrated-dataset-cofid
[3] https://www.langual.org/Default.asp

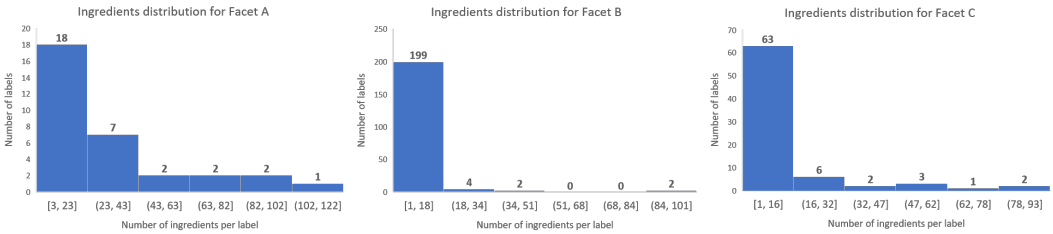


Figure 1 Class distribution histograms for the three facets.

Table 1 Number of class labels per facet and median number of ingredients per label.

Facet	Number of labels	Median of ingredients per label
A	32	14.5
B	207	2
C	77	4

The database of the nutritional recommender system contains 4970 ingredients, from which 902 are labelled according to all three considered facets. Ingredients distribution according to the three facets is depicted in Figure 1. This figure evinces an imbalance in class distribution. Facet A is the one with a less unbalanced number of ingredients, while Facet B presents a large number of labels with less than 10 ingredients, similar to what happens in Facet C. Facets B and C have a few major classes that engulf a lot of ingredients. Furthermore, the total number of labels per facet and the median of the number of ingredients for the labels of each facet is shown in Table 1.

In order to create proper embeddings, two paths could be taken: provide annotated data to the model, such as tuples of a query entity and its database matches; or learn unsupervised embeddings from the relationships between the entities. Still and all, given the available data, which is scarce and not well structured, none of the aforementioned paths is an ideal option for good performance. As a matter of fact, some of the groups of labels have only one ingredient. This will not generalize well for new data, requiring an alternative method. As mentioned in Section 1, retrofitting was used to create a set of pre-trained embeddings.

People often use hypernyms to express their preferences towards broad groups of ingredients, instead of referring to individual ones. The combination of synsets and lexicon databases, such as *WordNet* or *ConceptNet* [28], with the information provided by *LanguaL* facets on each ingredient creates an interesting set of features to classify ingredients. Following the work of Wu et al. [33], the approach designed for this task is based on using or creating embeddings for ingredient classification. The logic behind this method is that any entity can be embedded by a neural embedding model by learning feature representations for relationships among collections of those entities. The vector space is the same for all entities, which enables the model to rank entities, documents, or objects according to the similarity measure to a given query entity.

Bearing this in mind, each group of *LanguaL* facets can be regarded as a set of labels to classify each ingredient. The hypothesis we seek to explore in this work is that a correct classification of the ingredient in each of these three groups may enhance the retrieval of possible candidates from the ingredient database.

4 Pre-trained Embeddings

On a first stage, general pre-trained embeddings, such as *Word2Vec* and *GloVe*, were used to map every ingredient and *LanguaL* facets into a common vector space and hence, enabling us to classify each ingredient in the database as one of the *LanguaL* facets, for each one of the groups. This means that each ingredient was classified under three different groups of labels, one for each facet group.

Each one of the ingredient embedding, as well as each facet name embedding, was created by averaging the embeddings of each word that compose them, excluding stop-words. For that, several steps of pre-processing are required:

1. Tokenization – In order to obtain each individual token, spaCy tokenizer was used
2. Stop-words removal – Each extracted token is compared with a dictionary of stop-words so that they are removed to prevent retrofitting with them
3. Word normalization – Words are normalized in order to remove plural inflections which is particularly common in this dataset
4. Dictionary matching – The last step concerns matching the tokens with the words in the embeddings lexicon to extract the embeddings. However, there are some words that are not present in the lexicon or are not in the correct format. To solve this, the following steps are performed:
   a. Try to match bigrams with the lexicon; for hyphenated tokens, try matching their transformation into bigrams or unigrams (either by concatenating both tokens or by matching them individually).
   b. Perform fuzzy matching using the *FuzzyWuzzy* Python library, which calculates the Levenshtein distance to all possible words in the lexicon and retrieves the most probable candidate that has a distance of 1, if there is any.
   c. If not, create a zero-valued vector to emulate the embedding.

These steps were designed after exploring the database and finding some patterns in its data. For example, regarding the group *Fish and Seafood* (one of the labels for *LanguaL*'s Facet A group), the resulting embedding would be the average of the vectors for *Fish* and *Seafood*. This process is similar for ingredients, that usually have more than one token per name (see the examples in Section 6). Each ingredient and each class label is represented by only one vector, regardless of the number of tokens that compose its name. Class prediction is based on the Cosine Similarity between the embedding of a given ingredient and the embedding of each *LanguaL* class label, for each one of the three facets considered, following the work of Wu et al. [33]. The most similar class, for each facet, is the prediction made by the model. At the end of the classification process, each ingredient has 3 labels: one for each facet. Classification accuracy was used to evaluate the results obtained.

Word2Vec or *GloVe* are generic embeddings trained on large text corpora, and do not encode information extracted from knowledge graphs. We hypothesize that using embeddings that do, such as ConceptNet Numberbatch [28], is a sensible approach to understand whether the semantic information present in knowledge graphs can be leveraged to better enrich the embeddings in this particular context of food terms.

Table 2 summarizes the results obtained for two sets of pre-trained embeddings: GloVe (which incorporates no knowledge graph information) and ConceptNet Numberbatch (retrofitted with knowledge graph information).

As it is possible to note, the results obtained by using Numberbatch embeddings are higher than the ones obtained using GloVe. This shows that retrofitting does incorporate some semantic information into the embeddings, that leads to better semantic relationships between the different ingredients' embeddings.

5 Embeddings Refinement with Retrofitting

Being *LanguaL* an ontology which incorporates lexical information about food in a knowledge graph style, these established relations may be used to retrofit the pre-trained Numberbatch's embeddings so that they become more aware of food semantics. Thus, as a second step, the Numberbatch embeddings were retrofitted with the information that is available in *LanguaL*.

There are three possible ways to perform retrofitting with the available data:

A. Retrofitting class embeddings with the vectors retrieved for each of the ingredients that have that class as a descriptor in the database.
B. Retrofitting ingredient embeddings using the vectors of the classes that they have as descriptors in the database.
C. Combining the two previous approaches and retrofit both the ingredients and the classes.

Each procedure presents advantages and disadvantages. Procedure **A** will not change the embeddings associated with each ingredient. Instead, it encodes each *LanguaL* class according to the ingredients that it contains. This may pose as an advantage for real world applications when dealing with ingredients that are not part of the database. In these cases, provided that there are similar ingredients in the database, the model would still be able to classify the query. On the other hand, procedure **B** changes the vectors that represent the ingredients in order to match the *LanguaL* class label embedding. It is expected that this method will converge easily since each ingredient will only be retrofitted using, at the most, 3 concepts, which does not happen in the former method. Finally, procedure **C** tries to change both elements towards a converging representation. This will adapt both data types to each other which may be harmful when handling new information. This simultaneous change may cause the model to not generalise well for new ingredients as well as for new class labels, in case they are added to the database. Also, convergence may not be achieved since some of the concepts (ingredients and labels) are related in different ways. From iteration to iteration, the embeddings are being changed according to different embeddings (since both the ingredients and the class labels embeddings change). The selected approach should consider the performance obtained, after retrofitting, for all three facet groups at the same time, since there will only be one embedding representation per ingredient. Retrofitting each facet group individually and sequentially may harm the scores of the previously retrofitted facet groups. Even though one method may provide better results than another for an individual facet, the joint performance in the three facets should be considered. The results were obtained using k-fold cross validation with 6 randomly selected folds, (see Table 3). The folds could not be stratified since there are three simultaneous classification problems being addressed. Also, this means that some folds might not have data for every class in each of the three facets. The retrofitting session lasted for 20 iterations, after each of which the results were validated using the cosine similarity criterion, to consider updating the β

Table 2 Accuracy results for ingredient classification according to *LanguaL* facets, using different pre-trained embeddings models.

Model	Accuracy per facet		
	A	B	C
GloVe 400k vocabulary	0.277	0.324	0.087
ConceptNet Numberbatch	**0.417**	**0.401**	**0.089**

Table 3 Accuracy results for ingredient classification according to *LanguaL* facets, using three different retrofitting procedures.

Procedure	Accuracy		
	Facet A	Facet B	Facet C
Baseline results (no retrofitting)	0.412 ± 0.044	0.447 ± 0.027	0.108 ± 0.023
Approach A	$\mathbf{0.648 \pm 0.029}$	$\mathbf{0.505 \pm 0.050}$	$\mathbf{0.402 \pm 0.046}$
Approach B	0.412 ± 0.044	0.447 ± 0.027	0.108 ± 0.023
Approach C	0.504 ± 0.054	0.417 ± 0.050	0.242 ± 0.035

parameter (see Equation 2). If the results increased overall (throughout the 3 facets), β is maintained for the next iteration; otherwise, its value is increased by a factor of 20%. The initial β value is set to 1, the same as α; however, the latter is fixed for the whole session. At the starting point, both the vector to be retrofitted and the definitions have the same weight, which allows not to lose intrinsic information about the words, since the concept to be retrofitted should still retain some semantic meaning in order to prevent overfitting and generalize better to new data.

The results were obtained by evaluating the accuracy using 6-fold cross validation, since it was the maximum number that allowed to have all classes represented in each fold. By looking at Table 3 it is clear that the procedure that produces the best overall results is procedure A, where the classes were retrofitted incorporating information about the ingredients. This proved the hypothesis that procedure A deals better with unseen data than the other procedures, by not altering the values of the ingredient embeddings. During the training of procedure B it was not possible to make the model converge as the global accuracy was not stabilizing in a value, instead it was increasing and decreasing around the base value. This makes sense when thinking about the testing process. The ingredient embeddings of the training set are being altered according to the *LanguaL* information; however, the ones in the validation set have not suffered this alteration. This means that the classification score will be mostly the same in this case. Regarding procedure C, it is possible to see an improvement in both facets A and C, although the accuracy in facet B decreases. Bearing in mind these results and the perceived good handling of new data, procedure A was selected as the method to create the new set of embeddings that will be used in ingredient retrieval.

6 Token TF-IDF Weighting

Despite the fact that there is a clear improvement in classification accuracy, due to the naming format of the ingredients, the embeddings may be considering information that is not relevant for classification. The following list illustrates some examples of ingredient names present in the ingredients database:

- Pineapple, canned in juice
- Eggs, chicken, whole, raw
- Onions, raw
- Tuna, canned in brine, drained
- Peppers, capsicum, green, boiled in salted water

In most cases, ingredient names include extra information that may not be relevant for classification, such as the cooking method or the way of preservation. Moreover, the order of the words in the ingredients´ names does not always follow the same logic because the

ingredients derive from several sources. As a consequence, it was not possible to create rules for name processing before retrofitting. An example of such a rule would be to remove every word after the first comma; however, as is noticeable in the examples above, some important characteristics regarding ingredients are present after the first or even second comma. This pre-processing would have to be hand-made, which would be impractical. As a way to deal with this problem, a weighting mechanism based on the Term Frequency - Inverse Document Frequency (TF-IDF) weighting was applied to the tokens during ingredient and class label embedding creation. This way, words that are not important to distinguish classes will have reduced importance in the embeddings for retrofitting. TF-IDF weights were calculated in three different ways, depending on what was considered a document:

1. Concatenating ingredient names for a given class label
   In order to illustrate this case, when retrofitting the class label *Fish and Seafood* a document would comprise all ingredient names that belong to that class. This would boost the Term-Frequency part since there are usually several ingredients with similar names, varying only the cooking method, for example. Also, this would punish words that appear in different concepts, such as the cooking or preservation methods that are common to different types.
2. Considering each individual ingredient/class name
   In this case, Term-Frequency will not benefit, although Inverse Document Frequency will punish even more the tokens that appear in many ingredient or class names
3. Hybrid approach: the Term Frequency is calculated through procedure 1 while the Inverse Document Frequency is calculated using procedure 2
   With this hybrid approach the goal is to further punish words that appear in many classes while boosting words that belong to only one class.

Embeddings were retrofitted once again using method A with each of the TD-IDF approaches. We present and analyze the results obtained for the retrofitting procedure A, which gave the best results for the ingredient classification task under the three facet groups, as shown in Table 3, with an extra step of token weighting before constructing the ingredient or class label embedding. The results are presented in Table 4. Once again, they were taken using 6-fold cross validation. We can observe that TF-IDF weighting improves the results. Every experimented method increased performance when comparing to retrofitting without weighting. However, it is clear that TF-IDF 3, the hybrid method, presents the best overall results, showing the largest improvements in all three groups of labels. The differences between TF-IDF 2 and 3 have for all three facets have statistical significance. However, between TF-IDF 1 and 3 there is no statistical significance. TF-IDF 1 presented very similar results for facet B and facet C, although the error values are higher and the accuracy for facet A is lower. TF-IDF 2 presented the worst results of the three. Even though the differences between TF-IDF 1 and TF-IDF 3 have not statistical significance, the hybrid weighting method was slightly better at dealing with this kind of data due to selectively punishing the terms according to their frequency in different groups.

The evident improvement to using generic pre-trained embeddings shows that we were successful in incorporating food semantic information available in *LanguaL* into word vectors. The next section explores the algorithm designed to classify and retrieve ingredients based on a query entity, which takes advantage of a set of pre-trained ConceptNet Numberbatch embeddings retrofitted with *LanguaL* semantic information, through approach A and using the hybrid TF-IDF weighting (TF-IDF 3).

Table 4 Accuracy results for ingredient classification according to *LanguaL* facets, using approach A for retrofitting with different methods of TF-IDF weighting.

TF-IDF approach	Accuracy		
	Facet A	Facet B	Facet C
Baseline results (no TF-IDF weighting)	0.6475 ± 0.029	0.505 ± 0.050	0.402 ± 0.046
TF-IDF 1	0.690 ± 0.031	0.595 ± 0.064	0.451 ± 0.031
TF-IDF 2	0.682 ± 0.041	0.563 ± 0.061	0.411 ± 0.047
TF-IDF 3 (hybrid)	$\mathbf{0.692 \pm 0.026}$	$\mathbf{0.595 \pm 0.048}$	$\mathbf{0.451 \pm 0.021}$

Algorithm 1 Matching and Extracting Algorithm.

```
1  Input: Sentence, th1, th2, th3, th4
   Result: List of database ingredients that match the query entity
2  Ingred = getNamedEntity(Sentence)
3  IngredEmbedding = getEmbedding(Ingred)
4  [classA, classB, classC] = classifyIngred(IngredEmbedding)
5  EmbSimList, FuzMatchList = extractPossibleMatches([classA, classB, classC], th1,
     th2)
6  CandidateList = EmbSimList ⋂ FuzMatchList
7  if length(CandidateList) > 1 then
8  |   probableGroupMatch = calculateGroupMatchEmb(IngredEmb, classA, classB,
   |     classC, th3)
9  |   groupMatch = calculateGroupMatchLev(Ingred, probableGroupMatch, th4)
10 |   if length(groupMatch) > 0 then
11 |   |   Return extractFromDB(groupMatch)
12 |   else
13 |   |   Return CandidateList
14 else if length(CandidateList) == 1 then
15 |   Return CandidateList
16 else
17 |   Return []
```

7 Food Matching

The purpose of creating word embeddings that capture the semantic relations present in the *LanguaL* ontology was to retrieve, from the recommender system's ingredient database, the relevant ingredients, given a query entity extracted from user input. A perfect retrieval process would gather all ingredients that correspond to the Named Entity query based on the classification of the words (ingredients) that compose it. The entity may point to a group of ingredients, to a specific ingredient, or even to a group of ingredients that do not match *LanguaL* labels exactly. Bearing this in mind, a matching and extraction algorithm was developed. This algorithm leverages the food information that was incorporated in the embeddings and is detailed as Algorithm 1.

The first step is the Named Entity Recognition (line 2), in order to identify the ingredient that is present in the user's query. This entity, which represents a name of an ingredient, is then preprocessed, as described in Section 4, and encoded into an embedding (line 3). The next step is the classification of the query according to the three facets (line 4). Using

the predicted class labels, the algorithm extracts all ingredients that match at least 2 of the 3 classes from the database, resulting in a first list of possible candidates. Moreover, this list is then filtered using two criteria: embeddings cosine similarity and fuzzy matching to create two lists of ingredients that are strong candidates (line 5). Both these filters require predefined thresholds, th1 and th2, respectively. These previously mentioned lists are intersected in order to remove options from the fuzzy match search since it produces a large list with some unrelated ingredients. The result is the final list of candidates that will be the input to a series of conditions that will define the final result (line 6).

In case the Candidates List has only one ingredient, this ingredient is regarded as the match to the query and is returned by the system (lines 14 and 15). If the Candidates List has no elements, an empty array is returned, which means that there are no matches in the database for the ingredient query (lines 16 and 17). On the other hand, if the Candidates List has several elements, there is a strong possibility that the query is referring to a group of ingredients, rather than to a single ingredient. The query is then compared to the predicted class labels in a sequential process that uses cosine similarity and Levenshtein distance, requiring two threshold values: th3 and th4 respectively). The result of this last process (lines 8 and 9) is a list of *LanguaL* labels that may match the query. If this list is not empty, then the algorithm extracts all ingredients that are labeled accordingly (giving priority to facet A, then B and lastly C) (line 11). Otherwise, the Candidates List is returned. This means that the query matches a group of ingredients that is not a specific *LanguaL* group.

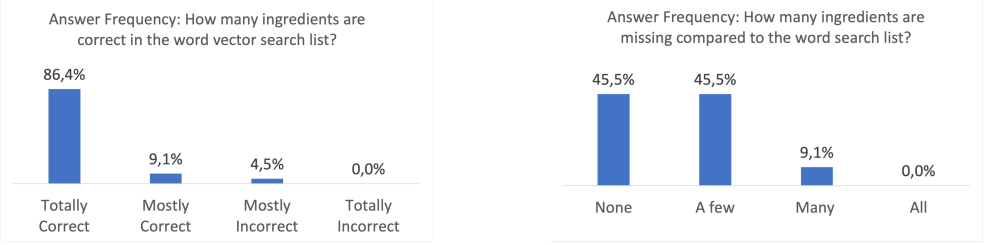
The thresholds referred in the algorithm were defined through observation, in order to maximize accuracy. An increase in the value of the threshold would represent an increase in precision with a consequent decrease in recall, since it causes a decrease in the number of ingredients retrieved. It makes the model more certain about the ingredients its extracting with the drawback of maybe missing some correct ones. This method was validated by user testing, as explained in Section 7.1, since there is a lack of an annotated dataset that could serve as validation.

7.1 User Validation

Due to the lack of a proper dataset, the matching algorithm was qualitatively evaluated by volunteers. Each volunteer must suggest one ingredient that has not yet been selected by others. Both the results obtained through the ingredient retrieval algorithm (based on embeddings) and the ones obtained by using a fuzzy word matching algorithm are shown to the volunteer, who has to answer three questions. This last search method is regarded as baseline method and was the only implemented in the recommender system. It compares the strings of the ingredients character by character.

1. Whether or not the ingredients retrieved by the embeddings-based algorithm are correct, on a 4-valued scale, from “totally incorrect” to “totally correct”.
2. Whether or not there is any ingredient obtained from word matching search that should also be in the embeddings results list, on a 4-valued scale, from “none” to “all”. It is worth noticing that the word matching search results usually contain ingredients whose name is similar to the one in the query, even though they do not match (e.g., *cheesecake* is a result of searching for *cheese*).
3. What is the preferred option for ingredient retrieval.

This way it is possible to evaluate the developed algorithm from a user standpoint, using some approximate recall and precision metrics. The test was performed by 22 volunteers (one ingredient each). Figure 2a shows the answer distribution for the question that addressed the



(a) Response frequency for food matching precision assessment.

(b) Response frequency for food matching recall assessment.

Figure 2 Qualitative results from the user ingredient retrieval evaluation.

correctness of the shown results by the embeddings-based search algorithm. These results can be mapped to the precision of the system, which measures how many of the positive answers are in fact true. Results show that the developed algorithm has a high level of precision. The large majority of the query outputs are totally correct, meaning that the ingredients the system shows are in fact related to the query term.

Recall is also another important metric that should be taken into account. Figure 2b shows the answers regarding the comparison made between the word-based search and the developed algorithm. The goal was to identify items that were correctly present in the former and missing in the latter. This does not calculate the true recall of the model, which would require a list of all correct items per query. Nonetheless, it is a useful comparison to detect missing items. The results show that the large majority of the answers were positive, meaning that the possible recall is also high. However, it is possible to affirm that the model shows increased precision when compared to recall. According to the answers that judged the preferred algorithm, the embeddings based algorithm explained in this paper was preferred by **77.3%** of the users. Even though the recall is lower than precision and the algorithm does not always gather all ingredient samples from the dataset, users prefer to have access to correct ingredients.

8 Conclusions

This work described the incorporation of semantic information from a food-related ontology into word embeddings, hence creating a set of embeddings that really capture the relations between food terms. Pre-trained embeddings were shown to poorly encode the different linkages that exist between food terms, creating the necessity of more semantic-aware embeddings. These relations are hard to capture with text data. Retrofitting has shown to be a valuable technique that enabled the enrichment of general knowledge embeddings in terms of food relations, largely increasing class similarity between the descriptor labels from *LanguaL* and the ingredients names. Also, from the three methods that were tried for retrofitting, regarding what embeddings to change (the class labels or the ingredients), the one that provided the best results was to alter the embeddings from the labels, based on the embeddings from the ingredients. This means that even if new ingredients are to be classified, this method should still be able to classify it correctly, leading to better results in the cross-validation testing. Moreover, TF-IDF weighting in the embedding creation proved to improve the results by giving different importance to the tokens that compose each ingredient and class name. This way, only words that are really specific and distinctive from each name are used to perform the retrofitting of the embeddings. TF-IDF was calculated in

three different ways, from which the one that gave the best results was a hybrid approach where the TF and IDF parts were calculated using different concepts of "Document". The implementation of this weighting mechanism allowed for an uptick in the class prediction accuracy. Further validation of this method and the resulting embeddings may be made to properly fine tune the embeddings. A validation data set may be created to analyze the methods and establish benchmark scores for several parameters such as measuring ingredient similarity through embeddings and comparing it to real life similarity. Even though the evaluation presented above showed very promising results, an exact and quantitative method should also be applied in order to further validate and reinforce these conclusions.

This work resulted in a set of pre-trained embeddings that already incorporate food knowledge and can be used for several applications besides database extraction and ingredient classification. These embeddings may pose as a useful tool for nutrition recommender systems or health companions in functionalities such as ingredient substitution or recipe creation by leveraging the ingredient relations and similarities. Another example of a possible application is the classification of recipes based on the ingredients that compose them, which then can be used to generate meal suggestions fostered by the similarity between user preferred meals and new ones. The applications that can be powered by these embeddings should also be properly validated by the creation of benchmark tests and scores. Moreover, this work evinces that the application of retrofitting as a way of enriching embeddings can be applied to virtually any context that requires grasping semantic relations, as long as there is a knowledge graph or similar structure that encodes these relations to support it.

References

1 Alaa A. Abd-alrazaq, Mohannad Alajlani, Ali Abdallah Alalwan, Bridgette M. Bewick, Peter Gardner, and Mowafa Househ. An overview of the features of chatbots in mental health: A scoping review, December 2019.

2 Andreas Arens-Volland, Benjamin Gateau, and Yannick Naudet. Semantic Modeling for Personalized Dietary Recommendation. *Proceedings - 13th International Workshop on Semantic and Social Media Adaptation and Personalization, SMAP 2018*, pages 93–98, 2018. doi:10.1109/SMAP.2018.8501864.

3 Ram G. Athreya, Axel Cyrille Ngonga Ngomo, and Ricardo Usbeck. Enhancing Community Interactions with Data-Driven Chatbots - The DBpedia Chatbot. In *The Web Conference 2018 - Companion of the World Wide Web Conference, WWW 2018*, pages 143–146, New York, New York, USA, April 2018. Association for Computing Machinery, Inc. doi:10.1145/3184558.3186964.

4 Timothy W. Bickmore, Daniel Schulman, and Candace L. Sidner. A reusable framework for health counseling dialogue systems based on a behavioral medicine ontology. *Journal of Biomedical Informatics*, 44(2):183–197, April 2011. doi:10.1016/j.jbi.2010.12.006.

5 Kyungyong Chung and Roy C. Park. Chatbot-based heathcare service with a knowledge base for cloud computing. *Cluster Computing*, 22(1):1925–1937, January 2019. doi:10.1007/s10586-018-2334-5.

6 Leigh Clark, Nadia Pantidi, Orla Cooney, Philip Doyle, Diego Garaialde, Justin Edwards, Brendan Spillane, Emer Gilmartin, Christine Murad, Cosmin Munteanu, Vincent Wade, and Benjamin R. Cowan. What makes a good conversation? Challenges in designing truly conversational agents. In *Conference on Human Factors in Computing Systems - Proceedings*, pages 1–12, New York, New York, USA, May 2019. Association for Computing Machinery. doi:10.1145/3290605.3300705.

7 Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, volume 1, pages 4171–4186, 2019. URL: https://github.com/tensorflow/tensor2tensor.

8 Vanesa Espín, María V. Hurtado, and Manuel Noguera. Nutrition for Elder Care: A nutritional semantic recommender system for the elderly. *Expert Systems*, 33(2):201–210, 2016. doi:10.1111/exsy.12143.

9 William J. Evans and Deanna Cyr-Campbell. Nutrition, exercise, and healthy aging. *Journal of the American Dietetic Association*, 97(6):632–638, 1997. doi:10.1016/S0002-8223(97)00160-0.

10 Manaal Faruqui, Jesse Dodge, Sujay K Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. Retrofitting word vectors to semantic lexicons. In *NAACL HLT 2015 - 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, pages 1606–1615, 2015. doi:10.3115/v1/n15-1184.

11 Catherine Havasi, Robert Speer, Kenneth Arnold, Henry Lieberman, Jason Alonso, and Jesse Moeller. Open mind common sense: Crowd-sourcing for common sense. In *AAAI Workshop - Technical Report*, volume WS-10-02, page 51, 2010. URL: www.aaai.org.

12 Shafquat Hussain, Omid Ameri Sianaki, and Nedal Ababneh. A Survey on Conversational Agents/Chatbots Classification and Design Techniques. In *Advances in Intelligent Systems and Computing*, volume 927, pages 946–956. Springer Verlag, 2019. doi:10.1007/978-3-030-15035-8{_}93.

13 H. N. Io and C. B. Lee. Chatbots and conversational agents: A bibliometric analysis. In *IEEE International Conference on Industrial Engineering and Engineering Management*, volume 2017-December, pages 215–219. IEEE Computer Society, February 2018. doi:10.1109/IEEM.2017.8289883.

14 J. D. Ireland and A. Møller. Langual food description: A learning process. *European Journal of Clinical Nutrition*, 64:S44–S48, 2010. doi:10.1038/ejcn.2010.209.

15 Veton Kepuska and Gamal Bohouta. Next-generation of virtual personal assistants (Microsoft Cortana, Apple Siri, Amazon Alexa and Google Home). In *2018 IEEE 8th Annual Computing and Communication Workshop and Conference, CCWC 2018*, volume 2018-January, pages 99–103. Institute of Electrical and Electronics Engineers Inc., February 2018. doi:10.1109/CCWC.2018.8301638.

16 Stefanie Mika. Challenges for nutrition recommender systems. *CEUR Workshop Proceedings*, 786:25–33, 2011.

17 Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*. International Conference on Learning Representations, ICLR, 2013.

18 George. Miller and Princeton University. Cognitive Science Laboratory. *WordNet*. MIT Press, 1998.

19 Jeffrey Pennington, Richard Socher, and Christopher D Manning. GloVe: Global vectors for word representation. In *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pages 1532–1543, 2014. doi:10.3115/v1/d14-1162.

20 Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, volume 1, pages 2227–2237. Association for Computational Linguistics (ACL), February 2018. doi:10.18653/v1/n18-1202.

21 Gorjan Popovski, Bibek Paudel, Tome Eftimov, and Barbara Korousic Seljak. Exploring a standardized language for describing foods using embedding techniques. In *Proceedings - 2019 IEEE International Conference on Big Data, Big Data 2019*, pages 5172–5176. Institute of Electrical and Electronics Engineers Inc., December 2019. doi:10.1109/BigData47090.2019.9005970.

22 David Ribeiro, João Machado, Jorge Ribeiro, Maria João M. Vasconcelos, Elsa F. Vieira, and Ana Correia De Barros. SousChef: Mobile meal recommender system for older adults. In *ICT4AWE 2017 - Proceedings of the 3rd International Conference on Information and Communication Technologies for Ageing Well and e-Health*, pages 36–45. SciTePress, 2017. doi:10.5220/0006281900360045.

23 David Ribeiro, Jorge Ribeiro, Maria João M. Vasconcelos, Elsa F. Vieira, and Ana Correia de Barros. SousChef: Improved meal recommender system for Portuguese older adults. *Communications in Computer and Information Science*, 869:107–126, 2018.

24 Christopher R Sauer and Alex Haigh. Cooking up Food Embeddings Understanding Flavors in the Recipe-Ingredient Graph, 2017.

25 Nuno Silva, David Ribeiro, and Liliana Ferreira. Information extraction from unstructured recipe data. *ACM International Conference Proceeding Series*, Part F1482:165–168, 2019. doi:10.1145/3323933.3324084.

26 Vivian S Silva, Andre Freitas, and Siegfried Handschuh. Building a knowledge graph from natural language definitions for interpretable text entailment recognition, 2018. URL: http://brat.nlplab.org/.

27 Vivian S Silva, Siegfried Handschuh, and Andre Freitas. Categorization of semantic roles for dictionary definitions, 2018. URL: https://www.aclweb.org/anthology/W16-5323.

28 Robyn Speer, Joshua Chin, and Catherine Havasi. ConceptNet 5.5: An Open Multilingual Graph of General Knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 4444–4451. AAAI Press, 2017.

29 Wesley Tansey, Edward W. Lowe, and James G. Scott. Diet2Vec: Multi-scale analysis of massive dietary data. *arXiv*, December 2016. arXiv:1612.00388.

30 Christoph Trattner and David Elsweiler. Food Recommender Systems Important Contributions, Challenges and Future Research Directions, November 2017. arXiv:1711.02760.

31 Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In I Guyon, U V Luxburg, S Bengio, H Wallach, R Fergus, S Vishwanathan, and R Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

32 Anu Venkatesh, Chandra Khatri, Ashwin Ram, Fenfei Guo, Raefer Gabriel, Ashish Nagar, Rohit Prasad, Ming Cheng, Behnam Hedayatnia, Angeliki Metallinou, Rahul Goel, Shaohua Yang, and Anirudh Raju. On Evaluating and Comparing Conversational Agents. In *Conversational AI Workshop at the 31st Conference on Neural Information Processing Systems*, pages 1–10, 2017. URL: http://alborz-geramifard.com/workshops/nips17-Conversational-AI/Papers/17nipsw-cai-evaluating_conversational.pdf.

33 Ledell Wu, Adam Fisch, Sumit Chopra, Keith Adams, Antoine Bordes, and Jason Weston. StarSpace: Embed all the things! In *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, pages 5569–5577. AAAI press, September 2018.

34 Anbang Xu, Zhe Liu, Yufan Guo, Vibha Sinha, and Rama Akkiraju. A new chatbot for customer service on social media. In *Conference on Human Factors in Computing Systems - Proceedings*, volume 2017-May, pages 3506–3510. Association for Computing Machinery, May 2017. doi:10.1145/3025453.3025496.

35 Wen Zhou, Haoshen Hong, Zihao Zhou, and Stanford Scpd. Derive Word Embeddings From Knowledge Graph, 2019.

TatWordNet: A Linguistic Linked Open Data-Integrated WordNet Resource for Tatar

Alexander Kirillovich
Kazan Federal University, Kazan, Russia
Higher School of Economics, Moscow, Russia

Marat Shaekhov
Kazan Federal University, Kazan, Russia

Alfiya Galieva
Kazan Federal University, Kazan, Russia

Olga Nevzorova
Kazan Federal University, Kazan, Russia
Higher School of Economics, Moscow, Russia

Dmitry Ilvovsky
Kazan Federal University, Kazan, Russia
Higher School of Economics, Moscow, Russia

Natalia Loukachevitch
Moscow State University, Moscow, Russia
Kazan Federal University, Kazan, Russia
Higher School of Economics, Moscow, Russia

Abstract

We present the first release of TatWordNet (`http://wordnet.tatar`), a wordnet resource for Tatar. TatWordNet has been constructed by the combination of the expand and the merge approaches. The synsets of TatWordNet have been compiled by: (i) the automatic conversion of concepts of TatThes, a socio-political Tatar; (ii) semi-automatic translation of synsets of RuWordNet, a wordnet resource for Russian with the followed manual verification and correction; (iii) manual translation of base RuWordNet synsets; (iv) and manual translation of the all hypernyms of the previously translated RuWordNet synsets. The currents version of TatWordNet contains 18,583 synsets, 36,540 lexical entries and 49,525 senses. The resource has been published to the Linguistic Linked Open Data cloud and interlinked with the Global WordNet Grid.

2012 ACM Subject Classification Computing methodologies → Language resources

Keywords and phrases Linguistic Linked Open Data, WordNet, Thesaurus, Tatar language

Digital Object Identifier 10.4230/OASIcs.LDK.2021.16

Supplementary Material *Dataset*: `http://wordnet.tatar/`

Funding The work was funded by Russian Science Foundation according to the research project no. 19-71-10056.

1 Introduction

The Princeton WordNet thesaurus (PWN) [9, 11] is one of the most important language resources for linguistic studies and natural language processing. PWN is a large-scale lexical knowledge base for English, organized as a semantic network of synsets. A synset is a set of words with the same part-of-speech that can be interchanged in several contexts. Synsets are interlinked by semantic relations, such as hyponymy (between specific and more general concepts), meronymy (between parts and wholes), antonymy (between opposite concepts) and other.

Inspired by success of PWN, many projects have been initiated to develop wordnets for other languages across the globe. Nowadays wordnet-like resources are developed for nearly 80 languages, but Tatar language is not among them. In this paper, we fill the gap and present the first release of TatWordNet (`http://wordnet.tatar`), a wordnet resource for Tatar.

2 Related works

At present time, there are various wordnets for some Turkic languages.

Two Turkish wordnet projects have been developed for the Turkish language. The first one [5, 2] has been created at Sabancı University as part of the BalkaNet project [19]. The BalkaNet project was built on the basis of a combination of expand and merge approaches. All wordnets contain many synonyms for Balkan common topics, as well as synsets typical for each of the BalkaNet languages. The size of Turkish Wordnet is about 15,000 synsets.

Another Turkish wordnet is the KeNet [1, 7, 8]. This wordnet was built on the basis of modern Turkish dictionaries. To build this resource, a bottom-up approach was used. Based on dictionaries, words were selected and then manually grouped into synsets. The relationships between words have been automatically extracted from dictionary definitions and then the latter have been fixed between synsets. The size of this resource is about 113,000 synsets.

Unfortunately, lack of large Turkish-Tatar dictionaries (as well as English-Tatar ones) makes it impossible to translate Turkish resources into the Tatar language. In this respect, the Tatar language can be attributed to low-resource languages.

The Extended Open Multilingual Wordnet [3] resource is built from Open Multilingual Wordnet by replenishing the WordNet data automatically extracted from the Wiktionary and Unicode Common Locale Data Repository (CLDR). The resource contains wordnets for 150 languages, including several Turkic: Azerbaijani, Kazakh, Kirghiz, Tatar, Turkmen, Turkish, and Uzbek. The Tatar wordnet contains a total of 550 concepts, which covers 5% of the PWN core concepts.

The BabelNet [18] resource contains a common network of concepts that have text inputs in many languages. The BabelNet contains 90,821 Tatar text entries that refer to 63,989 concepts. However, due to the fact that this resource was built automatically, it has quality issues. Thus, the development of a quality Tatar wordnet with an emphasis on the specific features of the Tatar language based on the existing lexical resources is very relevant.

3 TatWordNet construction

There are two main approaches for construction of wordnets for new languages: expand and merge [20]. The expand approach is to take the semantic network of PWN and translate its synsets into the target language, adding additional synsets when needed. The merge approach is to develop a semantic network in the target language from scratch and then link it to PWN.

Since the merge approach is very labor-intensive and time consuming, the expand approach seems more appropriate for under-resources languages such as Tatar. However, in development of TatWordNet, the expand approach can't be directly applied either, due to the lack of large English-Tatar dictionaries, necessary for translation of PWN to Tatar. At the same time, there are several relatively large and high-quality Russian-Tatar dictionaries, so Russian thesauri can be used as the source resources instead of PWN.

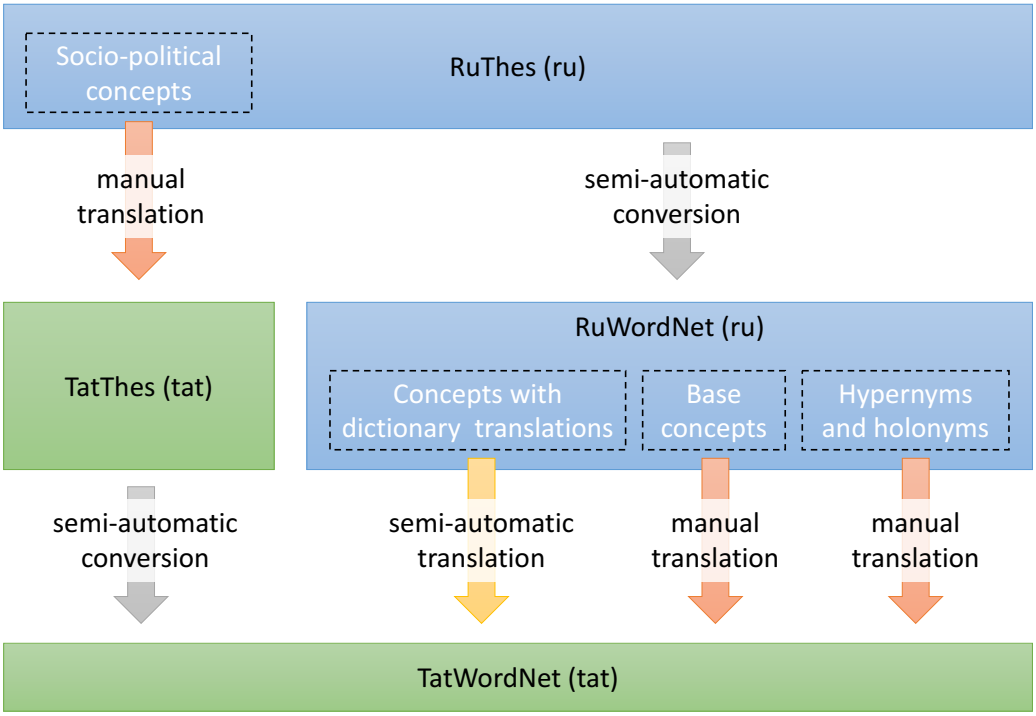


Figure 1 TatWordNet development.

With this consideration in mind we constructed TatWordNet on the base of three source resources, developed by us: RuThes, RuWordNet and TatThes. In this section we describe the resources, that were used to produce TatWordNet and then the construction process itself (Fig. 1).

RuThes

RuThes [16, 15, 13] is a thesaurus and a linguistic ontology for Russian. It is organized as a network of concepts, that are considered as language-independent "units of thought".

The concepts are language-independent in the sense that their identities and distinctions from each other don't depend on the terms that express them. At the same time, the network of concepts is linguistically motivated, i.e., it contains mostly those concepts, that are denoted by actual language expressions.

A concept is characterized by a unique name, and optionally by a gloss. Every concept is associated with lexical entries, by which this concept is referred to. The lexical entries associated with the same concept are called ontological synonyms. Ontological synonyms can comprise:

- words belonging to different parts of speech;
- language expressions relating to different linguistic styles and genres;
- idioms and free multiword expressions.

For example, the list of the lexical entries for the concept *Surgical operation* includes:

- the noun "операция" ("operation");
- the verbs "оперировать" ("to operate", imperfective) and "прооперировать" ("to operate", perfective);

- the adjectives “операционный” (“operative”) and “хирургический” (“surgical”);
- the noun phrases “хирургическая операция” (“surgical operation”), “хирургическое лечение” (“surgical treatment”) and “оперативное вмешательство” (“operative intervention”);
- the verb phrase “оперировать пациента” (“to operate on a patient”);
- and the idiom “лечь под нож” (“to go under the knife”).

Just like in WordNet, an ambiguous lexical entry is assigned to several concepts. For example, the word “коса” is assigned to three different concepts: *Tongue of land*, *Braid of hair* and *Scythe*.

RuThes defines four main relations between concepts:

1) The taxonomic relation, that is a union of the traditional ontological class-subclass (isA) and instanceOf relations. For example, this relation holds between the *Moscow* and the *City* concepts as well as between the *City* and *Settlement* concepts. The RuThes taxonomic relation is analogous to the hyponym-hypernym relation of WordNet.

2) The part-whole relation. In RuThes, this relation is interpreted fairly broadly, and applies to entities of many different types, including:

- physical objects (*Car engine* – *Car*), regions (*Europe* – *Eurasia*), substances, sets (*Battalion* – *Company*);
- processes (*Public prosecution* – *Judicial trial*);
- an attribute and its bearer (*Displacement* – *Ship*);
- a role or a participant of a situation and the situation (Teacher – Education);
- entities and situations in the encompassing sphere of activity (*Industrial plant* – *Industry*, *Tennis racket* – *Tennis*, *Tennis player* – *Tennis*).

At the same time the RuThes part-whole relation has a very important restriction: a concept-part should be related to its whole during normal existence of its instances. For example, the Tree concept is not described as part of the Forest concept, because trees can grow in many places, not only in forests. This makes it possible to use the transitivity of the part-whole relations with greater reliability.

The RuThes part-whole relation is similar to the WordNet meronym-holonym relation, and being applied to spheres of activity concepts it also can be used to model the WordNet domain relation.

3) The directed association relation, that expresses the relation of the external ontological dependence between two concepts. The association relation is established between two concepts C_1 and C_2 when C_1 ontologically depends on C_2 and C_1 is not a part of C_2. For example, this relation holds between the *Auto racing* and *Car* concepts.

4) The undirected symmetric association relation.

The RuThes relations have formal-ontological nature, allowing them to be subjected to the following formal inference rules: (1) transitivity of the part-whole relations; (2) inheritance of the part-whole relationships to subclasses; (3) inheritance of association relationships to parts and subclasses.

RuThes has considerable similarities with WordNet: both resources are composed of concepts/synsets, that are organized into a network by predefined set of conceptual relations, and associated with semantically related lexical entries.

At the same time, there are several differences between RuThes and WordNet, the most important of which is that a RuThes concept can be associated with lexical entries, belonging to different part of speech. Due to these differences, RuThes is not fully compatible with some WordNet-oriented NLP applications.

In order to obtain a Russian resource fully compatible with WordNet standards, the RuThes developers transformed RuThes to RuWordNet, a WordNet-like resource for Russian.

RuWordNet

RuWordNet (RWN) [14, 17] is a Russian wordnet, semi-automatically generated from the RuThes thesaurus.

To create RuWordNet, the single conceptual network of RuThes was transformed to synsets' subnets for each part of speech, which were then enriched by additional wordnet-specific relations.

This transformation was carried out by the following steps:

1) At the beginning, every RuThes concept was divided into part-of-speech-related synsets, in such a way that each synset contains all the lexical entries of the source concept which belong to the corresponding part-of-speech (i.e. a noun synset contains all the noun lexical entries of the source concept, a verb synset contains all the verbs, and the same for an adjective synset).

Fig. 1 represents examples of dividing a RuThes concept into part-of-speech-related RWN synsets. The *Coronation* concept has noun, verb and adjective lexical entries, and so it was divided into three part-of-speech-related synsets:

- the noun synset {"коронация" ("coronation")};
- the verb synset {"короновать" ("to crown"), "возвести на престол" ("to enthrone")};
- and the adjective synset {"коронационный" ("coronational")}.

At the same time, the Induction concept has only noun and verb lexical entries, and so it was divided into two synsets.

The synsets, obtained from the same source concept, were linked to each other with the relation of part-of-speech synonymy.

2) Then, the hyponym-hypernym and the meronym-holonym relations between RuThes concepts were reproduced for the corresponding RWN synsets of the same part-of-speech.

When two RuThes concepts, connected by the hyponym-hypernym relation, have corresponding RWN synsets of the same part-of-speech, the relation is established between these synsets.

It is, however, very common that in a pair of hyponym and hypernym concepts, one of the concepts does have a corresponding synset of a particular part-of-speech, but another concept doesn't. For example, the adjective synset {"коронационный" ("coronational")} is associated with the *Coronation* concept, but there is no adjective synset, associated with the *Coronation*'s hypernym, the *Induction* concept.

In such cases, hyponym-hypernym relation is established between two RWN synsets of the particular part-of-speech, if their source RuThes concepts are connected indirectly by the hyponym-hypernym path (and the intermediate concept in the path don't have themselves corresponding synsets of the relevant part of speech). For example, the relation is established between the aforementioned adjective synset {"коронационный" ("coronational")} and the adjective synset {"церемониальный" ("ceremonial")}, because their source RuThes concepts, *Coronation* and *Ceremony* respectively, are connected indirectly by a hyponym-hypernym path via the intermediate *Induction* concept (see Fig. 2).

In accordance with the WordNet standards, the hyponym-hypernym relation was additionally subdivided to the proper hyponym-hypernym and the instance hyponym-hypernym relations. In the current version of RuWordNet, the instance hyponym-hypernym relation was established for geographical objects.

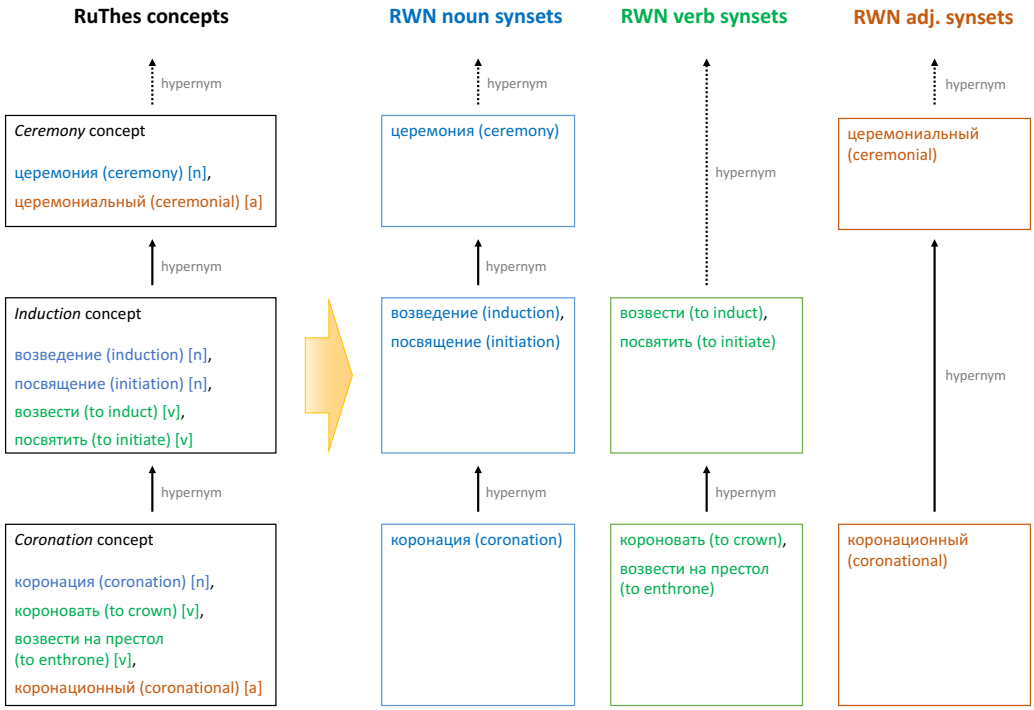


Figure 2 Transformation of the RuThes conceptual network into three RWN subnets of synsets (fragment).

The meronym-holonym relation was reproduced for the RWN synsets in the same way as the hyponym-hypernym one. After that, it was semi-automatically corrected according to the WordNet standards.

3) Finally, the established relations were semi-automatically supplemented by several wordnet-specific relations, including the antonymy relation, the relations of causation and entailment, the domain relation, the relations of word derivation and the relations between phrases and their components. This process is not relevant to the TatWordNet development, and so we will not describe it in this review.

4) Additionally, RuWordNet was linked to Princeton WordNet via the Global WordNet inter-lingual-index (ILI).

TatThes

TatThes [10] is a socio-political thesaurus for Tatar, developed on the basis of the conceptual network of the RuThes thesaurus.

TatThes can be described as a kind of satellite resource for RuThes: it doesn't define its own conceptual network, but heavily reuses the conceptual network of RuThes, extending it by new Tatar-specific concepts and supplementing the existing RuThes concepts by Tatar lexical entries. (It should be noted, that the reused RuThes concepts are defined only once in the RuThes itself, and TatThes only refers to them in accordance to the Linked Open Data principles).

Table 1 TatWordNet construction statistics.

Step	# of synsets
TatThes automatic conversion	4,422
RWN semi-automatic translation	13,366
RWN base concepts manual translation	135
Hypernyms manual translation	3,661

The development of TatThes was carried out by the following ways: 1) Supplementing an existing RuThes concept with Tatar lexical entries. Due to the language-independent nature of the RuThes conceptual network, it is mostly reused in Tatar thesaurus, even though the RuThes concepts can be expressed in Russian and Tatar texts in very different ways. For example, the *Age of majority* concept is expressed in Russian by the one noun "совершеннолетие", but in Tatar it is expressed by three verb phrases "буйга җитү", "яше җитү" and "балигъ булу".

A reused RuThes concept is supplemented by the Tatar translation of the concept name and by Tatar lexical entries.

The Russian and the Tatar lexical entries of the same concept can be described as cross-lingual ontological synonyms.

2) Adding a new hyponym concept and its lexical entries. The RuThes conceptual network can lack the concepts, specific to socio-cultural life of the Tatar society, such as Islam-related notions, social hierarchy of Oriental societies, Tatar ethno-cultural phenomena, etc. Such the concepts were added to TatThes as hyponyms of the existing RuThes concepts. For example, the *Muslim holiday* concept was added as a narrower concept of the *Holiday* concept.

3) Adding a new intermediate concept and its lexical entries. Even though the RuThes conceptual network is language-independent, it is nevertheless linguistically-motivated and thus can lack the concepts, lexicalized in Tatar, but not lexicalized in Russian. Many of such the concepts were added to TatThes on the intermediate level of the conceptual network, i.e. as a hyponym of one concepts and as a hypernym of the another. For example, RuThes contains *Stepson* and *Stepdaughter* concepts, but doesn't contain the concept of *Stepchild*. This concept was added to TatThes as a hyponym concept of the Relative and a hypernym of the *Stepson* and *Stepdaughter* concepts.

TatThes has been published on the Linguistic Linked Open Data cloud as part of RuThes Cloud project [12].

TatWordNet

With the described resources in hands, we constructed TatWordNet by the following steps: 1) Semi-automatic conversion of TatThes concepts to TatWordNet synsets. This conversion was performed in the same way as conversion of RuThes to RuWordNet. 2) Semi-automatic translation of the RuWordNet concepts. At first, we automatically translated the lexical entries by the Ganiev bilingual Russian-Tatar dictionary. Then we manually filtered out the incorrect translations, adding correct variants where necessary. 3) Manual translation of the Base RuWordNet concepts. 4) Manual translation of the hypernyms and holonyms of the previously translated RuWordNet concepts.

The number of TatWordNet synsets, obtained on each step is represented at Table 1.

4 TatWordNet description

TatWordNet is organized as networks of synsets, where each synset is linked to its lexical entries via lexical senses. The resource is distributed under the Creative Commons Attribution-ShareAlike License.

Linked Open Data representation

TatWordNet has been published to the Linguistic Linked Open Data cloud [6] and interlinked with the Global WordNet Grid [21] via the Collaborative Interlingual Index [4].

The resource is represented in terms of Global WordNet ontology as well as Onto-Lex/Lemon, SKOS, LexInfo and PROV ontologies.

Listing 1 represents the *City* synset and one of its lexical entries, senses and synset relations.

Listing 1 The *City* synset, its lexical entries, senses and relations.

```
@prefix skos: <http://www.w3.org/2004/02/skos/core#>.
@prefix wn: <https://globalwordnet.github.io/schemas/wn#>.
@prefix ontolex: <http://www.w3.org/ns/lemon/ontolex#>.
@prefix lexinfo: <http://www.lexinfo.net/ontology/2.0/lexinfo#>.
@prefix prov: <http://www.w3.org/ns/prov#>.

#Synset City
<http://lod.wordnet.tatar/synset/242-N>
  a ontolex:LexicalConcept, skos:Concept;
  wn:partOfSpeech wn:noun;
  skos:altLabel "шәһәр"@tt, "кала"@tt;
  ontolex:isEvokedBy
    <http://lod.wordnet.tatar/entry/шәһәр>,
    <http://lod.wordnet.tatar/entry/кала>;
  ontolex:lexicalizedSense
    <http://lod.wordnet.tatar/sense/242-N-шәһәр>,
    <http://lod.wordnet.tatar/sense/242-N-кала>;
  wn:domain_topic
    <http://lod.wordnet.tatar/synset/1702-N>;
  wn:hypernym
    <http://lod.wordnet.tatar/synset/123680-N>,
    <http://lod.wordnet.tatar/synset/145516-N>;
  wn:hyponym
    <http://lod.wordnet.tatar/synset/207-N>,
    <http://lod.wordnet.tatar/synset/3208-N>,
    <...>;
  wn:mero_part
    <http://lod.wordnet.tatar/synset/9171-N>,
    <http://lod.wordnet.tatar/synset/4250-N>,
    <http://lod.wordnet.tatar/synset/4773-N>;
  skos:inScheme
    <http://lod.wordnet.tatar/tatwordnet>;
  prov:wasGeneratedBy
    <http://lod.wordnet.tatar/prov/ganiev-translation>.

#Synset relation
<http://lod.wordnet.tatar/relation/hypernym-from-242-N-to-123680-N>
```

```
  a vartrans:LexicoSemanticRelation ;
  vartrans:category wn:hypernym;
  vartrans:source
    <http://lod.wordnet.tatar/synset/242-N>;
  vartrans:target
    <http://lod.wordnet.tatar/synset/123680-N>.

#Lexical sense, linking the concept to its lexical entry
<http://lod.wordnet.tatar/sense/242-N-шәһәр>
  a ontolex:LexicalSense ;
  ontolex:reference
    <http://lod.wordnet.tatar/synset/242-N>;
  ontolex:isLexicalizedSenseOf
    <http://lod.wordnet.tatar/synset/242-N>;
  ontolex:isSenseOf
    <http://lod.wordnet.tatar/entry/шәһәр>.

#Lexical entry
<http://lod.wordnet.tatar/entry/шәһәр>
  a ontolex:LexicalEntry, ontolex:Word;
  rdfs:label "шәһәр"@tt;
  wn:partOfSpeech wn:noun;
  lexinfo:partOfSpeech lexinfo:noun;
  ontolex:canonicalForm
    [
    rdf:type ontolex:Form;
    ontolex:writtenRep "шәһәр"@tt;
    ];
  ontolex:evokes
    <http://lod.wordnet.tatar/synset/242-N>;
  ontolex:sense
    <http://lod.wordnet.tatar/sense/242-N-шәһәр>;
  prov:wasGeneratedBy
    <http://lod.wordnet.tatar/prov/ganiev-translation>.
```

This representation is quite straightforward and mainly reflects GWA recommendations. The only three comments should be made: (1) In order to achieve computability with standard SKOS applications, the lexical entries were additionally represented as SKOS labels. (2) In accordance with OWL punning, the synset relationships were represented as RDF object properties and at the same time as RDF individuals. (3) Each synset was provided by the `prov:wasGeneratedBy` link to the method used to produce this synset (i.e. automatic generation from a TatThes concept, manual translation of a RuWordNet synset, etc).

Publishing on the Web

TatWordNet has been published on the Web (`http://wordnet.tatar/`) and is available via:

- dereferenceable URIs: `http://lod.wordnet.tatar`;
- SPARQL endpoint: `http://lod.wordnet.tatar/sparql`;
- RDF dump: `http://wordnet.tatar/download/twn.ttl.zip`.

Access to the resource via dereferenceable URIs is supported by mechanisms of content negotiation. When a web browser requests a URI, it is redirected to a web page with an HTML view of the entity, but the Semantic Web agent request is redirected to the page with the RDF representation.

Table 2 TatWordNet statistics.

Entity type	Count
Synset	18,538
Lexical entry	36,540
Word	13,469
Multi-word expression	23,071
Lexical sense	49,525
Synset relation	68,558
hypernym / hyponym	24,740
instance hypernym / hyponym	221
part holonym / meronym	1,336
domain topic	15,964
Link to inter-lingual index	3,661

Statistics

Statistics of RuThes Cloud is represented at Table 2.

SPARQL query example

Integration of TatWordNet to the LLOD cloud makes it possible to construct very complex federated SPARQL queries. Example of such queries is the following: find the Russian sentences, containing the words whose Tatar translations are hyponyms of the given Tatar word “шәһәр” (“city”) (Listing 2). This query utilizes several types of links: (1) between Russian corpora OpenCorpora and RuThes thesaurus, (2) cross-lingual links between RuThes and TatWordNet, and (3) finally internal links between TatWordNet synsets.

Listing 2 SPARQL query example.

```
PREFIX wn: <http://globalwordnet.github.io/schemas/wn#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX ontolex: <http://www.w3.org/ns/lemon/ontolex#>
PREFIX nif: <http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core#>
PREFIX conll: <http://ufal.mff.cuni.cz/conll2009-st/task-description.html#>

SELECT ?twn_hyponym ?opencorpora_word_text ?opencorpora_sentence_text {
    "шәһәр"@tt
        ^ontolex:writtenRep /
        ^ontolex:canonicalForm /
        ^ontolex:isEvokedBy ?twn_synset.
    ?twn_synset (wn:hyponym | wn:instance_hyponym)* ?twn_hyponym.
    ?twn_hyponym skos:closeMatch ?ruthes_concept.

    SERVICE <http://data.llod.ru/repositories/ruthes-cloud> {
        ?ruthes_concept
            ontolex:lexicalizedSense /
            ontolex:isSenseOf ?ruthes_entry.

        SERVICE <http://data.llod.ru/repositories/opencorpora> {
            ?ruthes_entry ^conll:LEXICAL_ENTRY_URI ?opencorpora_word.
            ?opencorpora_word nif:sentence ?opencorpora_sentence.
            ?opencorpora_word nif:anchorOf ?opencorpora_word_text.
            ?opencorpora_sentence nif:anchorOf ?opencorpora_sentence_text.
        }
    }
}
```

The query results contain 113 sentences, for example “Российскую столицу впервые посетил известнейший художник-визионёр Алекс Грей” (“The famous painter Alex Grey made his first visit to the Russian capital”).

5 Conclusion

In this paper, we present the first release of TatWordNet (`http://wordnet.tatar`), a Linguistic Linked Open Data-integrated wordnet resource for Tatar. TatWordNet was constructed on the base of three source resources, developed by us: RuThes, RuWordNet and TatThes. The currents version of TatWordNet contains 18,583 synsets, 36,540 lexical entries and 49,525 senses. The resource has been published to the Linguistic Linked Open Data cloud and interlinked with the Global WordNet Grid.

References

1 Özge Bakay, Özlem Ergelen, Elif Sarmış, Selin Yıldırım, Atilla Kocabalcıoglu, Bilge Nas Arıcan, Merve Özçelik, Ezgi Sanıyar, Oguzhan Kuyrukcu, Begüm Avar, and Olcay Yildiz. Turkish WordNet KeNet. In Piek Vossen and Christiane Fellbaum, editors, *Proceedings of the 11th Global Wordnet Conference (GWC 2021), Potchefstroom, South Africa, 18–21 Jan, 2021*, pages 166–174. GWA, 2021. URL: `https://www.aclweb.org/anthology/2021.gwc-1.19/`.

2 Orhan Bilgin, Özlem Çetinoğlu, and Kemal Oflazer. Building a Wordnet for Turkish. *Romanian Journal of Information Science and Technology*, 7(1–2):163–172, 2004. URL: `http://research.sabanciuniv.edu/379/`.

3 Francis Bond and Ryan Foster. Linking and Extending an Open Multilingual Wordnet. In Hinrich Schuetze et al., editors, *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013), Sofia, Bulgaria, 4–9 August, 2013. Volume 1: Long Papers*, pages 1352–1362. ACL, 2013. URL: `https://www.aclweb.org/anthology/P13-1133`.

4 Francis Bond, Piek Vossen, John P. McCrae, and Christiane Fellbaum. CILI: The collaborative interlingual index. In Christiane Fellbaum, Piek Vossen, Verginica Barbu Mititelu, and Corina Forascu, editors, *Proceedings of the 8th Global WordNet Conference (GWC 2016), Bucharest, Romania, 27–30 January, 2016*, pages 50–57. GWA, 2016. URL: `https://www.aclweb.org/anthology/2016.gwc-1.9/`.

5 Özlem Çetinoğlu, Orhan Bilgin, and Kemal Oflazer. Turkish Wordnet. In Kemal Oflazer and Murat Saraçlar, editors, *Turkish Natural Language Processing*, pages 317–336. Springer, 2018. `doi:10.1007/978-3-319-90165-7_15`.

6 Philipp Cimiano, Christian Chiarcos, John P. McCrae, and Jorge Gracia. Linguistic Linked Open Data Cloud. In Philipp Cimiano et al., editors, *Linguistic Linked Data: Representation, Generation and Applications*, pages 29–41. Springer, 2020. `doi:10.1007/978-3-030-30225-2_3`.

7 Elin Ehsani. *KeNet: A Comprehensive Turkish Wordnet and Using it in Text Clustering*. PhD thesis, Işık University, 2018. `doi:10.13140/RG.2.2.20932.27524`.

8 Razieh Ehsani, Ercan Solak, and Olcay Taner Yildiz. Constructing a WordNet for Turkish Using Manual and Automatic Annotation. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 17(3), April 2018. `doi:10.1145/3185664`.

9 Christiane Fellbaum. Wordnet. In Roberto Poli et al., editors, *Theory and Applications of Ontology: Computer Applications*, pages 231–243. Springer, 2010. `doi:10.1007/978-90-481-8847-5_10`.

10 Alfiya Galieva, Alexander Kirillovich, Bulat Khakimov, Natalia Loukachevitch, Olga Nevzorova, and Dzhavdet Suleymanov. Toward Domain-Specific Russian-Tatar Thesaurus Construction. In Radomir Bolgov et al., editors, *Proceedings of the International Conference IMS-2017, St. Petersburg, Russia, 21–24 June 2017*, ACM International Conference Proceeding Series, pages 120–124. ACM Press, New York, 2017. `doi:10.1145/3143699.3143716`.

11 Adam Kilgarriff and Christiane Fellbaum. *WordNet: an Electronic Lexical Database*. MIT Press, 2000.

12 Alexander Kirillovich, Olga Nevzorova, Emil Gimadiev, and Natalia Loukachevitch. RuThes Cloud: Towards a Multilevel Linguistic Linked Open Data Resource for Russian. In Przemysław

Różewski and Christoph Lange, editors, *Proceedings of the 8th International Conference on Knowledge Engineering and Semantic Web (KESW 2017), Szczecin, Poland, November 8–10, 2017*, Communications in Computer and Information Science, vol. 786, pages 38–52. Springer, 2017. doi:10.1007/978-3-319-69548-8_4.

13 N. Loukachevitch, B. Dobrov, and I. Chetviorkin. RuThes-lite, a Publicly Available Version of Thesaurus of Russian Language RuThes. In *Computational Linguistics and Intellectual Technologies: Papers from the Annual International Conference «Dialogue»*, pages 340–349. RGGU, 2014. URL: http://www.dialog-21.ru/digests/dialog2014/materials/pdf/LoukachevitchNV.pdf.

14 N. V. Loukachevitch, G. Lashevich, A. A. Gerasimova, V. V. Ivanov, and B. V. Dobrov. Creating Russian WordNet by Conversion. In *Computational Linguistics and Intellectual Technologies: papers from the Annual Conference "Dialogue"*, pages 405–415. RGGU, 2016. URL: http://www.dialog-21.ru/media/3409/loukachevitchnvetal.pdf.

15 Natalia Loukachevitch and Boris Dobrov. RuThes Linguistic Ontology vs. Russian Wordnets. In Heili Orav, Christiane Fellbaum, and Piek Vossen, editors, *Proceedings of the 7th Global Wordnet Conference (GWC 2014), Tartu, Estonia, 25–29 January, 2014*, pages 154–162. University of Tartu Press, 2014. URL: https://www.aclweb.org/anthology/W14-0121/.

16 Natalia Loukachevitch and Boris Dobrov. RuThes Thesaurus for Natural Language Processing. In Daria Gritsenko, Mariëlle Wijermars, and Mikhail Kopotev, editors, *The Palgrave Handbook of Digital Russia Studies*, pages 319–334. Palgrave Macmillan, 2021. doi:10.1007/978-3-030-42855-6_18.

17 Natalia Loukachevitch, German Lashevich, and Boris Dobrov. Comparing two thesaurus representations for Russian. In Francis Bond, Takayuki Kuribayashi, Christiane Fellbaum, and Piek Vossen, editors, *Proceedings of the 9th Global Wordnet Conference (GWC 2018), Singapore, 8–12 January, 2018*, pages 34–43. GWA, 2018. URL: https://www.aclweb.org/anthology/2018.gwc-1.5/.

18 Roberto Navigli and Simone Paolo Ponzetto. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250, December 2012. doi:10.1016/j.artint.2012.07.001.

19 D. Tufis, D. Cristeau, and S. Stamou. BalkaNet: Aims, Methods, Results and Perspectives. A General Overview. *Romanian Journal of Information Science and Technology*, 7(1–2):9–43, 2004. URL: http://www.dblab.upatras.gr/balkanet/journal/7_Overview.pdf.

20 Piek Vossen. EuroWordNet General Document. Technical Report LE2-4003, LE4-8328, University of Amsterdam, July 1999. URL: http://www.illc.uva.nl/EuroWordNet/docs.html.

21 Piek Vossen, Francis Bond, and John P. McCrae. Toward a truly multilingual global wordnet grid. In Christiane Fellbaum, Piek Vossen, Verginica Barbu Mititelu, and Corina Forascu, editors, *Proceedings of the 8th Global WordNet Conference (GWC 2016), Bucharest, Romania, 27–30 January, 2016*, pages 419–426. GWA, 2016. URL: https://www.aclweb.org/anthology/2016.gwc-1.59/.

Explainable Zero-Shot Topic Extraction Using a Common-Sense Knowledge Graph

Ismail Harrando
EURECOM, Sophia Antipolis, Biot, France

Raphaël Troncy
EURECOM, Sophia Antipolis, Biot, France

Abstract

Pre-trained word embeddings constitute an essential building block for many NLP systems and applications, notably when labeled data is scarce. However, since they compress word meanings into a fixed-dimensional representation, their use usually lack interpretability beyond a measure of similarity and linear analogies that do not always reflect real-world word relatedness, which can be important for many NLP applications. In this paper, we propose a model which extracts topics from text documents based on the common-sense knowledge available in ConceptNet [24] – a semantic concept graph that explicitly encodes real-world relations between words – and without any human supervision. When combining both ConceptNet's knowledge graph and graph embeddings, our approach outperforms other baselines in the zero-shot setting, while generating a human-understandable explanation for its predictions through the knowledge graph. We study the importance of some modeling choices and criteria for designing the model, and we demonstrate that it can be used to label data for a supervised classifier to achieve an even better performance without relying on any humanly-annotated training data. We publish the code of our approach at `https://github.com/D2KLab/ZeSTE` and we provide a user friendly demo at `https://zeste.tools.eurecom.fr/`.

2012 ACM Subject Classification Computing methodologies → Information extraction

Keywords and phrases Topic Extraction, Zero-Shot Classification, Explainable NLP, Knowledge Graph

Digital Object Identifier 10.4230/OASIcs.LDK.2021.17

Supplementary Material *Software (Source Code)*: `http://github.com/D2KLab/ZeSTE`
archived at `swh:1:dir:1d6fc42dda71a72d6869ff6a2ba46c209e82cf07`
InteractiveResource (Website): `https://zeste.tools.eurecom.fr/`

Funding This work has been partially supported by the French National Research Agency (ANR) within the ASRAEL (ANR-15-CE23-0018) and ANTRACT (ANR-17-CE38-0010) projects, and by the European Union's Horizon 2020 research and innovation program within the MeMAD (GA 780069) and SILKNOW (GA 769504) projects.

1 Introduction

Word2Vec [14], GloVe [16], BERT [5] along with its many variants are among the most cited works in NLP. They have demonstrated the possibility of creating generic, cross-task, context-free and contextualized word representations from big volumes of unlabeled text, which can be then used to improve the performance of numerous down-stream NLP tasks by bringing free "real world knowledge" about words meanings and usage, learned mostly through word co-occurrences statistics, thus cutting down the need for substantial amounts of labeled data. However, being compacted representations of word meanings, these embeddings do not offer much in terms of interpretation: we know that similar words tend to have similar representations (i.e. similar orientation in the embedding space), and that some analogies can be found by doing linear algebraic operations in the embedding space (such as the

now-famous $v_{King} - v_{Man} + v_{Woman} \approx v_{Queen}$). Both measures, however, fall short when evaluated systematically, as there is an entire literature about studying the limits of analogies and the biases that these word embeddings can encode depending on the corpora they have been trained on [4, 2, 15, 13].

In this paper, we consider the task of *topic categorization*, a sub-task of text classification where the goal is to label a textual document such as a news article or a video transcript, into one of multiple predefined *topics*, i.e. labels that are related to the topical content of the document. Common examples for news topics are *"Politics"*, *"Sports"* and *"Business"*. What is interesting about this task, compared to other text classification tasks such as *spam detection* or *sentiment analysis*, is that the content of the document to classify is *semantically related* to the labels themselves, providing an interesting case for zero-shot prediction setting. Zero-shot prediction, broadly defined, is the task of predicting the class for some input without having been exposed to any labeled data from that class.

To do so, we propose to leverage *ConceptNet*, a knowledge graph that aims to model common sense knowledge into a computer- and human-readable formalism. Coupled with its graph embeddings (ConceptNet Numberbatch[1]), we show that using this resource does not only achieve better empirical results on the task of zero-shot topic categorization, but also does so in an explainable fashion. With every word being a node in the knowledge graph, it is straightforward to justify the similarity between words in the document and its assigned label, which is not possible for other distributional word embeddings as they are built on the statistical aggregations of large volumes of textual data.

The remainder of this paper is structured as follows: we present some related work for text categorization emphasizing the methods that make use of external semantic knowledge (Section 2). We present our proposed method, named **ZeSTE** (**Ze**ro **S**hot **T**opic **E**xtraction) in Section 3. We empirically evaluate our approach for zero-shot topic categorization in Section 4 where we compare it to different baselines on multiple topic categorization benchmark datasets (including a non-English dataset). We also test our method against a few-shot setup and show how our approach can be combined with a supervised classifier to obtain competitive results on the studied datasets without relying on any annotated data. In Section 5, we describe a demo that we developed that enable users to provide their own set of labels and observe the explanations for the model predictions. Finally, we conclude and outline some potential future improvements in Section 6.

2 Related Work

Nearly all recent state-of-the-art Text Categorization models ([29, 3, 28, 25], to cite a few) rely on some form of Transformer-based architecture [27], pre-trained on large text corpora. While the task of using fully-unsupervised, non-parametric models for text categorization is yet to be explored to the best of our knowledge, there has been multiple efforts to incorporate common-sense knowledge as a basis for many artificial intelligence tasks, especially in a zero-shot setting where humans seem to be able to satisfactorily perform a new task by relying mostly on their common sense and prior knowledge accumulated from their interaction with the world.

In this paper, we propose to leverage ConceptNet [24], a multilingual semantic graph containing statements about common-sense knowledge. The nodes represent concepts (words and phrases, e.g. `/c/en/sport`, `/c/en/belief_system`, `/c/en/ideology`, `/c/fr/coup_d'_état`) from 78 languages, linked together by semantic relations such as `/r/IsA`, `/r/RelatedTo`,

[1] `https://github.com/commonsense/conceptnet-numberbatch`

`/r/Synonym`, `/r/PartOf`. The graph contains over 8 million nodes and 21 million edges, expressed in triplets such as (`/c/en/president`, `/r/DefinedAs`, `/c/en/head_of_state`). It was built by aggregating facts from the Open Mind Common Sense project [20], parsing Wiktionary[2], Multilingual WordNet [8], OpenCyc [7], as well as a subset of DBpedia, and designed to explicitly express facts about the real world and the usage of words and concepts that is necessary to understand natural language. Along with the graph, *ConceptNet Numberbatch* are multilingual pre-trained word (and concept) embeddings that are built on top of the ConceptNet knowledge graph. They are generated by computing the Positive Pointwise Mutual Information (PPMI) for the matrix representation of the graph, reducing its dimensionality, and then using "expanded retrofitting" [23] to make them more robust and linguistically representative by combining them with Word2Vec and GloVe embeddings. While the approach can be carried using other linguistic resources such as WordNet [8], we choose to use ConceptNet because it models word relations that are more relevant to the task of Topic Categorization such as `/r/RelatedTo`, which is the most present relation in the graph.

[6] is an early example of leveraging semantic knowledge to improve text categorization. It uses the relations in WordNet [8] to enhance the Bag of Word representation of documents by mapping the different words from a document into their entries in WordNet, and adding those as well as their hypernyms to the Bag of Words count. This, followed by a statistical χ^2 test to reduce the dimension of the feature vector, leads to a significant improvement over the simple bag-of-word model. [21] introduces *Graph of Words*, in which every document is represented by a graph of its terms, all connected with relations reflecting the co-occurrence information (terms appearing within a window of size w are joined by an edge). The authors propose a weighting scheme for the traditional TF-IDF model, where nodes are weighted based on some graph centrality measure (degree, closeness, PageRank), and edges are weighted with Word2Vec word embedding cosine similarity between their nodes. Incorporating both graph structure and distributional semantics from the embeddings to compute a weight for each term yields significantly better results on multiple text classification datasets.

[30] benchmark the task of zero-shot text classification, underlining the lack of work reported on this challenge in the NLP community in comparison to the field of computer vision. They distinguish two definitions of zero-shot text categorization: *Restrictive*, in which during a training phase, the classifier is allowed to see a subset of the data with the corresponding labels, but during inference, it is tested on a new subset of examples from the same dataset but not pertaining to any of the seen labels; *Wild*, where the classifier is not allowed to see any examples from the labeled data but can use Wikipedia's categories as a proxy dataset, for example. Our method fits into this second definition, although it does not require any training data. The authors compare some methods in both regimes (restrictive and wild) and they propose "Entail", a model based on BERT [5] and trained on the task of textual entailment evaluated on the Yahoo! Comprehensive Questions and Answers dataset.

[17] tackle the task of zero-shot text classification by projecting both the document and the label into an embedding space and using multiple architectures to measure the relatedness of the document and label embeddings. At test time, the classifier is able to ingest labels that were not seen during the training phase, but share the same embedding space with the labels already seen. A similar approach is followed by [22], in which both documents and labels are embedded into a shared cross-lingual semantic representations (CLESA) built upon Wikipedia as a multilingual corpus, and then the prediction is made by measuring the similarity between the two representations.

[2] `https://en.wiktionary.org/wiki/Wiktionary:Main_Page`

Finally, [31] propose a two-stage framework for zero-shot document categorization, combining 4 kinds of semantic knowledge: distributional word embeddings, class descriptions, class hierarchy, and the ConceptNet knowledge graph. In the first phase, a (coarse-grained) classifier is trained to decide whether the document at hand comes from a class that was seen during the training phase or not. This is done by training one ConvNet classifier [11] per label in the "seen" dataset, and setting a confidence threshold that, if none of the classifiers meets, the document is considered to be for the unseen labels. Secondly, a fine-grained classifier predicts the document final label. If the document is from a "seen" label, then the corresponding pretrained ConvNet classifier is picked. Otherwise, a zero-shot classifier which takes as input a representation of the document, the label, and their ConceptNet closeness, is trained on the seen labels but is expected to generalize to unseen ones as they share the same embedding space.

3 Approach

Our approach aims to perform topic categorization without relying on any in-domain labeled or unlabeled examples. Our underlying assumption is that words belonging to a certain topic are part of a vocabulary that is semantically related to its humanly-selected candidate label, e.g. a document about the topic of *"Sports"* will likely mention words that are semantically related to the word *Sport* itself, such as *team*, *ball*, and *score*. We use ConceptNet [24] to produce a list of candidate words related to the labels we are interested in. We generate a "topic neighborhood" for each topic label which contains all the semantically related concepts/nodes, and we then compute a score for each label based on the document content. Figure 1 illustrates our approach using a simple example.

3.1 Generating Topic Neighborhoods

To generate the topic neighborhoods for a given label, we query ConceptNet for nodes that are directly connected to the label node. Since the number of calls to the online API is capped at 120 queries/minute, we instead use the dump[3] of all ConceptNet v5.7 assertions, keeping only the English and French concepts for the English and French datasets, resulting in 3,323,321 (resp. 2,943,446) triplets, respectively. Although the assertions contain a finer granularity when it comes to referring to concepts, we only consider the root word for each concept to build the neighborhood. For example, the word "match" has multiple meanings: the tool to light a fire `/c/en/match/n/wn/artifact`, the event where two contenders meet to play `/c/en/match/n/wn/event`, and the concept of several things fitting together `/c/en/match/n/wn/cognition`. All these nodes (as well as others such as the verb form) will be mapped to the same term: "match". We also add (inverse) relations from the object to the subject for each triplet to ensure that every term in the graph has a neighborhood. The total number of unique triplets is 6,412,966, with 1,165,189 unique nodes for English (6.413.002 and 1.448.297 for French, respectively).

The topic neighborhood is created by querying every node that is N hops away from the label node. Every node is then given a score that is based on the cosine similarity between the label and the node computed using *ConceptNet Numberbatch* (ConceptNet's graph embeddings). This score represents the relevance of any term in the neighborhood to the main label, and would also allow us to refine the neighborhood and produce a score. In the

[3] `https://github.com/commonsense/conceptnet5/wiki/Downloads#assertions`

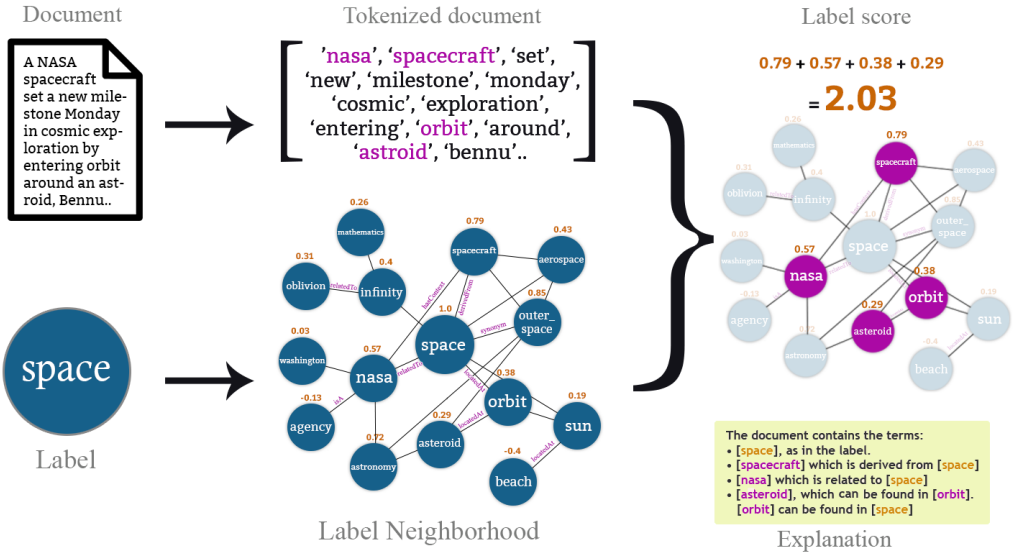


Figure 1 Illustration of ZeSTE: given a document and a label, we start by pre-processing and tokenizing the document into a list of terms, and we generate the label neighborhood graph by querying ConceptNet (we omit some relation labels in the figure for clarity). Each node on the graph is associated with a score that corresponds to the cosine similarity between the graph embeddings of that node and the label node. We use the overlap between the document terms and the label neighborhood to generate a score for the label, as well as an explanation for the prediction. After doing so for all candidate labels, we pick the one with the highest score to associate to the document at hand.

case of a label which has multiple tokens (e.g. the topic "Arts, Culture, and Entertainment"), we just take the union of all word components' neighborhoods, weighted by the maximum similarity score if the same concept appear in the vicinity of multiple label components.

The higher N is, and the bigger the generated neighborhoods become. We thus propose multiple methods to vary the size of the neighborhood:

1. **Coverage**: we vary the number of hops N;
2. **Relation masking**: we consider subsets of all possible relations between words from the ConcepNet knowledge graph. More precisely, we consider three cases:
   a. The sole relation *RelatedTo* which is the most frequent one in the graph;
   b. The 10 semantic and lexical *similarity* relations only, i.e. *'DefinedAs', 'DerivedFrom', 'HasA', 'InstanceOf', 'IsA', 'PartOf', 'RelatedTo', 'SimilarTo', 'Synonym', 'Antonym'*;
   c. The whole set of 47 relations defined in ConceptNet.
3. **Filtering**: we filter out some nodes based on their similarity score:
   a. Threshold (*Thresh* T): we only keep nodes in the neighborhood if their similarity score to the label node is greater than a given threshold T.
   b. Hard Cut *(Top* N*)*: we only keep the top N nodes in the neighborhood ranked by their similarity score.
   c. Soft Cut *(Top* $P\%$*)*: we only keep the top $P\%$ nodes in the neighborhood, ranked on their similarity score.

3.2 Scoring a Document

Once the neighborhood is generated, we can predict the document label by quantifying the overlap between the document content (as broken down to a list of tokens) and the label neighborhood nodes, which we denote in the following equations as $doc \cap LN(label)$. We consider the following scoring schemes:

1. **Counting**: assigning the document with the highest overlap count between its terms and the topic neighborhood.

$$count\_score(doc, label) = |doc \cap LN(label)| \tag{1}$$

2. **Distance**: factoring in the graph the distance between the term in the document and the label (number of nodes or path length between the token node and the label): the further a term is from the label vicinity, the lower is its contribution to the score.

$$distance\_score(doc, label) = \sum_{token \in doc \cap LN(label)} \frac{1}{min\_path\_length(token, label) + 1} \tag{2}$$

3. **Degree**: each node's score is computed using the number of incoming edges to it, reflecting its importance in the topic graph (we use $f(n) = log(1 + n_{edges})$ to amortize nodes with a very high degree).

$$degree\_score(doc, label) = \sum_{token \in doc \cap LN(label)} f(node\_degree(token)) \tag{3}$$

4. **Numberbatch similarity**: for each term in the document included in the label neighborhood, we increase the score by its similarity to the label embedding (we denote the Numberbatch concept embedding for word w by nb_w).

$$numberbatch\_score(doc, label) = \sum_{token \in doc \cap LN(label)} sim(nb_{token}, nb_{label}) \tag{4}$$

5. **Word Embedding similarity**: similar to the Numberbatch similarity, but we use pre-trained 300-dimensional GloVe [16] word embeddings instead to measure the word similarity (we denote the GloVe word embedding for word w by $glove_w$).

$$glove\_score(doc, label) = \sum_{token \in doc \cap LN(label)} sim(glove_{token}, glove_{label}) \tag{5}$$

We observe that in equations 4 and 5, multiple similarity measures and normalization options were considered, but the cosine similarity empirically showed the best results, so it has been used for the rest of the experiments. The model is thus the set of the neighborhood for each candidate label coupled with a scoring scheme. We discuss in Section 4.2 (Model Selection) how to empirically decide on the best filtering and scoring method that we then use in our experiments and our online demo.

3.3 Explainability

Given the label neighborhood, we can generate an explanation as to why a document has been given a specific label. This explanation can be generated in natural language or shown as the subgraph of ConceptNet that connects the label node and every word in the document

that appears within its neighborhood, and hence counted towards its score3.1. We note that, although the "RelatedTo" edge does not offer much in term of explanation beyond semantic relatedness, its explicit presence in ConceptNet confirms this relatedness beyond any non-explicit measure (e.g. word embedding similarity). Since this graph is usually quite big, we can generate a more manageable summary by picking up the closest N terms to the label in the graph embedding space, as they constitute the nodes contributing most to the score of the document. We can show one path (for instance, the shortest) between each of the top term nodes and the label node. The paths can then be verbalized in natural language. For example, for the label `Sport`, and a document containing the word *Stadium*, a line from the explanation (i.e. a path on the explanation subgraph) would look like this (`r/RelatedTo` and `r/IsA` are two relations from ConceptNet):

> The document contains the word "Stadium", which is *related to* "Baseball". "Baseball" *is a* "Sport".

Another method of explaining the predictions of the model is to highlight the words (or n-grams) that contributed to the classification score in the document. Since every word that appear both in the document and the label neighborhood has a similarity score associated to it (e.g. the cosine similarity between the word and the label embedding), we can visually highlight the words that are relevant to the topic. These two explanation methods are further discussed in the Section 5.

4 Experiments

In this section, we first describe the datasets which have been used to evaluate our approach (Section 4.1). Next, we present experiments to select the best model (Section 4.2). We then detail the zero-shot baselines that we compare to our approach (Section 4.3) before discussing our results (Section 4.4). Finally, we show how our model can be used to bootstrap the training for supervised classifier to achieve significantly better results (Section 4.5).

4.1 Datasets

While the premise of our approach is the possibility to perform topic categorization in a zero-shot setting, we evaluate it on several datasets from the literature. We identify 4 different Topic Categorization datasets with different properties in terms of style (professional news sources or user-generated content), size, number of topics, topic distribution and document length. We also evaluate our model on a new dataset named AFP News, which provides interesting comparison grounds such as multilingualism (available in English and French), multi-topical documents and strong imbalance in topics distribution. Table 3 summarizes the characteristics of each of these 5 datasets.

- **20 Newsgroups** [12]: a collection of 18000 user-generated forum posts arranged into 20 groups seen as topics such as *"Baseball", "Space", "Cryptography", and "Middle East"*.
- **AFP News** [18]: a dataset containing 125K English and 26K French news articles issued by the French News Agency (*Agence France Presse*). The articles are tagged with one or more topics coming from IPTC NewsCode taxonomy[4]. We consider the first level of this taxonomy which corresponds to 17 top-level topics such as *"Art, Culture and Entertainment", "Environment", or "Lifestyle and Leisure"*. The label distribution is

[4] `http://cv.iptc.org/newscodes/subjectcode/`

highly unbalanced. Since the data on both the English and French documents come from the same source and have similar properties, we use this dataset to compare how well our method compare on two different languages.

- **AG News** [10]: a news dataset containing 127600 English news articles from various sources. Articles are fairly distributed among 4 categories: *"World", "Sports", "Business" and "Sci/Tech"*.
- **BBC News** [9]: a news dataset from BBC containing 2225 English news articles classified in 5 categories: *"Politics", "Business", "Entertainment", "Sports" and "Tech"*.
- **Yahoo! Answers Comprehensive Dataset** [26]: a dataset containing over 4 million questions (title and body) and their answers submitted by users, extracted from the Yahoo! Answers website. We construct the evaluation dataset following the procedure described in [30] to reproduce its setup for comparison: we select 10K questions from each of the top 10 categories on Yahoo! Answers. We split it into 2 categories. The first split contains the labels *"Health", "Family & Relationships", "Business & Finance", "Computers and Internet" and "Society and Culture"* whereas the second split contains the labels *"Entertainment & Music", "Sports", "Science & Mathematics", "Education & Reference", and "Politics & Government"*. The ground-truth topic labels are assigned by users.

In order to determine the filtering criteria as discussed in Section 4.2 without relying on any further dataset-specific tuning, we use the BBC News dataset as a development set to select the optimal parameters for our model, under the hypothesis that the properties that work best for this dataset would work best for others as well. We verify post-hoc that this hypothesis holds empirically, i.e., the design choices decided using BBC News turn out to deliver the best results on the other datasets as well. The filtering criteria values that gave the best results for *Threshold*, *Hard Cut* and *Soft Cut* have empirically been set to $T = 0.0$, $N = 20000$, $P = 50\%$, respectively.

The 5 datasets have all been pre-processed using the same procedure: we lowercase the text, remove all non-alphabetical symbols and English (or French) stopwords. We then tokenize the strings using the space as separator and finally lemmatize the word using `WordNetLemmatizer`[5]. If the dataset has multiple textual contents (e.g. the Yahoo! Questions dataset consists of questions that are made of a title, a question body, and a set of answers), we concatenate them to form one "document". In the case of the AFP News dataset, each document can be tagged with one label, multiple labels, or no labels. We drop all non-tagged documents. To compute accuracy, we consider a prediction to be correct if it is among the document labels, and false otherwise. Finally, for the 20 Newsgroups dataset, we collapse the categories "comp.os.ms-windows.misc" and "comp.windows.x" into "windows", and "comp.sys.mac.hardware" and "comp.sys.ibm.pc.hardware" into "hardware", since they have very similar original labels. We do so for the baselines methods as well.

4.2 Model Selection

In this section, we evaluate some of the options regarding the neighborhood filtering and document scoring mentioned in Section 3. We use the *BBC News* dataset as a testbed for evaluating model selection. We report the results on the other datasets using the best parameters found at this stage. We first evaluate the different choices made to generate the label neighborhood as discussed in Section 3.1 and reported in Table 1.

[5] `http://www.nltk.org/api/nltk.stem.html?highlight=lemmatizer#module-nltk.stem.wordnet`

Table 1 Comparing the different filtering configurations on the BBC News dataset (performance expressed in Accuracy).

		Filtering method			
Relations	Depth	Keep All	Top50%	Top20K	Thresh
One	N = 1	55.4	54.5	55.4	55.4
	N = 2	69.0	65.8	64.8	66.2
	N = 3	81.0	81.3	83.5	81.3
Similarity	N = 1	60.8	57.5	60.8	60.8
	N = 2	70.3	66.9	66.2	68.0
	N = 3	77.9	81.9	83.4	81.9
All	N = 1	68.4	674	68.4	68.4
	N = 2	75.2	73.8	78.0	73.9
	N = 3	83.6	83.6	84.0	83.6

We observe that the most consistent way of improving the results is to use larger neighborhoods, as 3-hops neighborhoods systematically outperform the 1 and 2-hops ones. Our experiments show that going beyond $N = 3$ comes at the cost of increasing the computation time (mainly the computation of cosine similarity between the label and related nodes), while offering only very marginal improvement overall. The filtering method also impacts the performance but not as consistently (especially for $N = 3$). Finally, using all the relations generally yields better results than using only a subset of the relations, enough to justify the speed trade-off. It is also worth noting that using only the "r/RelatedTo" relation yields comparatively good results, which highlights the fact that "common-sense word relatedness" as expressed in ConceptNet is a strong signal for topic categorization.

For the scoring scheme, we evaluate the various methods mentioned in Section 3.2. The results are reported in Table 2.

Table 2 Evaluating the scoring schemes on BBC News (performance expressed in Accuracy).

Count	Distance	Degree	Numberbatch	GloVe
81.8	77.8	78.1	84.0	81.6

We see that using the ConceptNet Numberbatch embeddings gives the best result as they can condense the count, distance, degree of the nodes and the linguistic similarity with regard to the label into a measure of similarity in the embedding space. Accounting for term frequency (counting a word twice in the scoring if it appears twice in the document) in all of the scoring schemes did not translate to an improvement on the results. Accounting for n-grams, however, seems to slightly improve the results, but they require the availability of a corpus to mine such n-grams. Therefore, for the rest of our experiments, we do not account for n-grams. For the rest of our experiments, we keep the following configuration: *("All relations", N = 3, "Top20K", "Numberbatch scoring")*. We use ConceptNet v5.7 and Numberbatch embeddings v19.08.

Table 3 Performance on five Topic Categorization datasets (Accuracy).

Dataset	BBC News	AG News	20 Newsgroups	AFP News (FR)	YQA-v0	YQA-v1
# topics	5	4	20	17	5	5
# docs	2225	127600	18000	125516	50000	50000
doc/topic std	54.3	22.4	56.7	13682.7	0.0	0.0
Avg.words/doc	390	40	122	242	43	44
EN	26.1	26.7	53.5	60.0	51.8	36.2
GWA	40.2	63.9	36.7	32.8	49.9	43.4
Entail [30]	71.1	64.0	45.8	61.8	52.0	49.3
ZeSTE	**84.0**	**72.0**	**63.0**	**80.9** (**78.2**)	**60.3**	**58.4**
Supervised	96.4	95.5	88.5		72.6	80.6
Method	[19]	[29]	[28]		[30]	

4.3 Baselines

We propose 3 baseline systems:

- *Entail*: this model is provided by HuggingFace[6] [30]. We use `bart-large-mnli` as our backend Transformer model which can also be tested at `https://huggingface.co/zero-shot/`.
- *GloVe Weighted Average* (GWA) inspired by [1]: we average the 300-d GloVe embeddings vectors for every word in the document, and use the cosine similarity between the document embedding and the GloVe label embedding as a score to classify the document. For multi-worded labels (e.g. "Middle East"), we use the average vector of all the label components as the label embedding.
- *Embedding Neighborhood* (EN): for each label, we select the 20k closest words in the embedding space. We score each document by adding up the cosine similarity between the GloVe embedding of every word in the document that appears in the "embedding neighborhood" and the GloVe embedding of the label. In other words, we substitute the explicit graph connections in ConceptNet with the closeness in the GloVe embedding space. This baseline reflects the ability of generic embeddings to encode the topicality of words based only on the similarity in the embedding space.

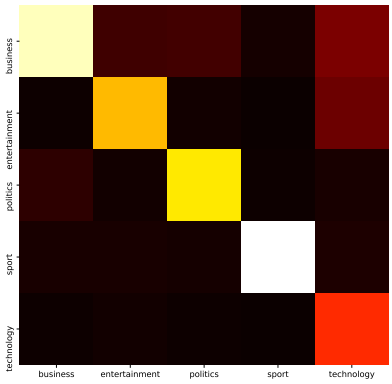
4.4 Zero-Shot Results

We provide the results obtained by evaluating our method against the baselines on the 5 datasets (BBC News, AG News, 20 Newsgroups, AFP News and YQA) in Table 3. Our method surpasses both GloVe baselines with a significant margin in accuracy on all datasets. `GWA` shows that the generic word embeddings poorly encode the topicality of words, as it is based solely on the similarity scores between the document content and the label world embedding. The low results with `EN` show that filtering based only on the embedding space (instead of the graph) is insufficient since the rarely-used words tend to clutter the embedding neighborhood. `ZeSTE` significantly outperforms `Entail`, despite the fact that the later relies on a large corpus pre-training and *textual entailment* task fine-tuning.

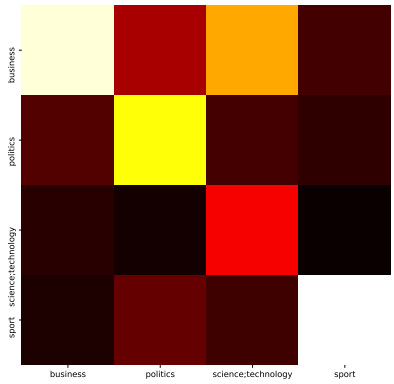
The confusion matrices for each datasets (Figure 2) indicate that our method performs more poorly on datasets where there is a lot of topical overlap between the different labels. For example, on 20 Newsgroups, "alt.atheism", "soc.religion.christian", "talk.religion.misc"

[6] We are using the implementation provided at `https://github.com/katanaml/sample-apps/tree/master/01`

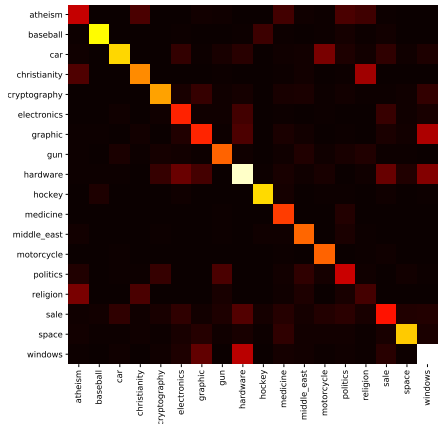
have a lot of overlapping vocabulary, leading to most documents under "alt.atheism" to fall into either other options. If we collapse all three labels into one (e.g. "religion"), the performance improves from 63.0% to 68.9%. We also observe on the AFP News dataset that "politics" intersects with "unrest, conflict, war" and "business, finance". The lack of a diameter pattern in AFP's confusion matrix is due to the high imbalance in the labels, which hurts the precision of the model. It is also worth mentioning how the method works seamlessly for other languages, as demonstrated on the French AFP News dataset, which sees a slight drop of accuracy from 80.9% on English to 78.2% accuracy on French. This shows a great potential for multilingual applicability as ConceptNet supports 78 languages.



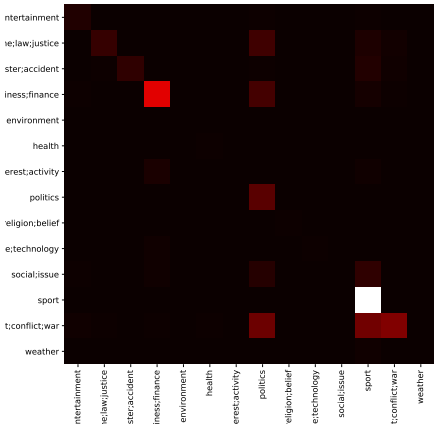
(a) BBC News.



(b) AG News.



(c) 20 NewsGroup.



(d) AFP News.

Figure 2 Confusion Matrices for the 4 news datasets.

Our method is clearly outperformed by the fully supervised methods. While the drop in performance is significant for some datasets, it is to be observed that the supervised methods not only rely on the availability of labeled training data, but usually also require expensive pre-training on more data. For instance, [29] use XLNet, an autoregressive Transformer that has been pre-trained on 120 GB of text. We consider that this absolute loss of accuracy performance is counter-balanced by the applicability in a zero-shot setting as well as the explainability of the model's decision.

Finally, we note that the choice of the initial label can be critical for the functioning of this method. While we stayed true to the original labels in the experiments (with an exception for the label "World" that was replaced with "news, politics" in the AG News dataset), we are aware of the possibility of obtaining even better results by changing a label to a more fitting one or including more keywords into it.

4.5 Few-Shots Setup

For each dataset, we compare our model to a more realistic use-case. We create a 80-20 training/test split if one is not already provided, and we randomly sample n examples from each category to create a training set for our supervised classifier. Among the classifiers considered, we find uncased BERT (*BertForSequenceClassification*) to perform the best. We grow n in increments of 10 until we achieve an empirical accuracy score on the test set that surpasses our approach in the zero-shot setting. We report $N = n * |labels|$ the number of documents that need to be annotated in Table 4. We also observe that increasing the number of documents does not always improve the test set accuracy.

Table 4 The required number of documents needed to achieve zero-shot best performance.

Dataset	BBC News	AG News	20 Newsgroups	AFP News
N	300	240	2160	8500

4.6 Bootstrapping a Supervised Classifier

One of the potential usage of zero-shot classification is to provide "automatic labeling" for unlabeled documents to a traditional supervised classifier. In other words, we use ZeSTE to annotate a portion of each dataset, and we feed these annotated examples to a state-of-the-art text classifier.

We first define the confidence of the classification as the normalized score for each label, i.e. divided by the sum of all candidate labels scores. In Figure 3, which shows the error distribution with respect to the classification confidence, we see that it correlates well with whether the label is correct or not. Therefore, we can use it as a signal to pick samples to use to bootstrap our classifier. We train the same few-shots model from 4.5 on the best 60% examples of our training data, i.e. we drop 40% of the training examples on which ZeSTE is least confident. We report on the results in Table 5 (the results for ZeSTE row correspond to the performance on the test-set only, not the entire dataset as in Table 3). We can clearly see how the bootstrapping process helps the classifier achieving significantly better results on all tested datasets, all without requiring any human annotation. It is worth mentioning that for this application, the BERT-based classifier training was not thoroughly fine-tuned, which means that even better results can be achieved using the same automatic labeling setup.

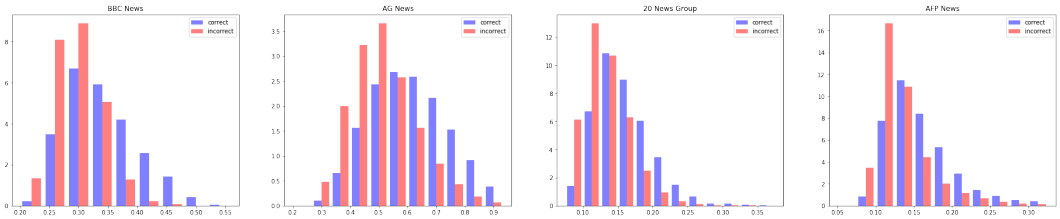


Figure 3 The prediction error distribution along the normalized confidence scores.

Table 5 The accuracy of ZeSTE and used as bootstrapped model (using the generated predictions as training data) on the test split of each dataset.

Dataset	BBC News	AG News	20 Newsgroups	AFP News
ZeSTE	80.6	71.0	61.6	73.8
ZeSTE + BERT	**94.3**	**84.2**	**70.1**	**83.0**

5 Online Demo

To demonstrate our method, we developed a web application which allows users to create their own topic classifier in real time. The user inputs the text to classify either by typing it into the designated textbox or by providing the URI of a web document that we scrape for extracting the content using Trafilatura[7]. The user is then prompted to either choose one of the pre-defined sets of labels (e.g. 20NG or IPTC used to evaluate the AFP dataset), or to provide her own set of label candidates. Once the user clicks on the "Predict the Topics" button, the server computes and caches the label neighborhood if it is the first time it encounters the label, otherwise it loads it from the cache for near real-time topic inference. Once the document is pre-processed and the label neighborhood generated, the server sends back its predictions (as confidence scores for each label candidate), and an explanation for each topic based on the common-sense connections between the document content and the label is provided (Figure 4, right panel). We only sample one path between document terms and the label, when in reality there could be many, in order to have a usable UI. In the future, we aim to depict the explanation as a subgraph of ConceptNet which shows all the relevant terms and their connections in the label neighborhood. We also highlight the relevant words in the input text (based on their score). While the demo works only for textual document written in English, we expect to support other languages in the future. The user interface makes use of the ZeSTE API which we also expose for others to be easily integrated.

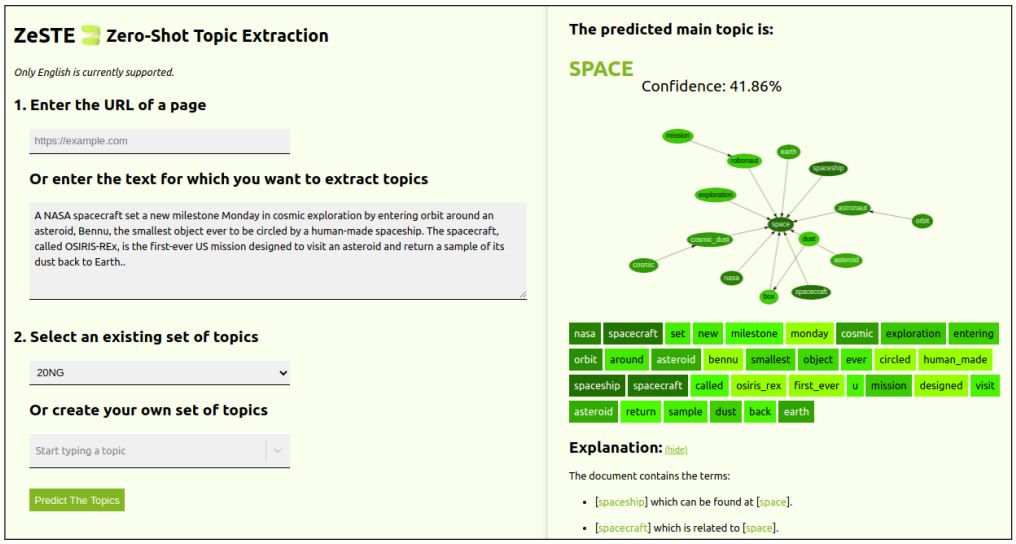


Figure 4 ZeSTE's User Interface deployed at `https://zeste.tools.eurecom.fr/`.

[7] `https://pypi.org/project/trafilatura/`

6 Conclusion and Future Work

In this work, we present ZeSTE, a novel method for zero-shot topic categorization that achieves competitive performance for this task, outperforming solid baselines and previous works while not requiring any labeled data. Our method also provides explainable predictions using the common-sense knowledge contained in ConceptNet. We demonstrate that ZeSTE can help to bootstrap a supervised classifier, achieving high accuracy on all datasets without requiring human supervision. The code to reproduce our approach and replicate our results is available at `https://github.com/D2KLab/ZeSTE`.

As an extension to this work, we consider an adaptation of the approach to other NLP tasks such as multi-class topic categorization, query expansion and keyphrase extraction. To further improve the approach, an analysis on how to partition the topic neighborhoods and minimise overlap is also envisaged. Finally, studying how to automatically pick better topic labels based on measures such as Mutual Information and Graph Centrality is to follow.

References

1 Katherine Bailey and Sunny Chopra. Few-shot text classification with pre-trained word embeddings and a human in the loop. arXiv, 2018. `arXiv:1804.02063`.

2 Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in neural information processing systems*, pages 4349–4357, 2016.

3 Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. Universal Sentence Encoder. arXiv, 2018. `arXiv:1803.11175`.

4 Dawn Chen, Joshua C Peterson, and Thomas L Griffiths. Evaluating vector-space models of analogy. arXiv, 2017. `arXiv:1705.04416`.

5 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pages 4171–4186. Association for Computational Linguistics, 2019.

6 Zakaria Elberrichi, Abdelattif Rahmoun, and Mohamed Amine Bentaalah. Using WordNet for Text Categorization. *International Arab Journal of Information Technology (IAJIT)*, 5(1), 2008.

7 Charles Elkan and Russell Greiner. Building large knowledge-based systems: Representation and inference in the Cyc project. *Artificial Intelligence*, 61(1):41–52, 1993.

8 Ingo Feinerer and Kurt Hornik. *wordnet: WordNet Interface*, 2017. R package version 0.1-14. URL: `https://CRAN.R-project.org/package=wordnet`.

9 Derek Greene and Pádraig Cunningham. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *23^{rd} International Conference on Machine learning (ICML)*, pages 377–384, 2006.

10 Antonio Gulli. *AG's corpus of news articles*, 2005. URL: `http://groups.di.unipi.it/~gulli/AG_corpus_of_news_articles.html`.

11 Yoon Kim. Convolutional neural networks for sentence classification. arXiv, 2014. `arXiv:1408.5882`.

12 Ken Lang. Newsweeder: Learning to filter netnews. In *12^{th} International Conference on Machine Learning (ICML)*, pages 331–339, 1995.

13 Chandler May, Alex Wang, Shikha Bordia, Samuel R. Bowman, and Rachel Rudinger. On Measuring Social Biases in Sentence Encoders. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pages 622–628. Association for Computational Linguistics, 2019.

14 Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

15 Orestis Papakyriakopoulos, Simon Hegelich, Juan Carlos Medina Serrano, and Fabienne Marco. Bias in Word Embeddings. In *International Conference on Fairness, Accountability and Transparency (FAT)*, pages 446—457. Association for Computing Machinery, 2020.

16 Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *International Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

17 Pushpankar Kumar Pushp and Muktabh Mayank Srivastava. Train once, test anywhere: Zero-shot learning for text classification. arXiv, 2017. `arXiv:1712.05972`.

18 Charlotte Rudnik, Thibault Ehrhart, Olivier Ferret, Denis Teyssou, Raphaël Troncy, and Xavier Tannier. Searching News Articles Using an Event Knowledge Graph Leveraged by Wikidata. In *5th Wiki Workshop*, pages 1232–1239, 2019.

19 Vishal S Shirsat, Rajkumar S Jagdale, and Sachin N Deshmukh. Sentence level sentiment identification and calculation from news articles using machine learning techniques. In *Computing, Communication and Signal Processing*, pages 371–376. Springer, 2019.

20 Push Singh, Thomas Lin, Erik T Mueller, Grace Lim, Travell Perkins, and Wan Li Zhu. Open mind common sense: Knowledge acquisition from the general public. In *OTM Confederated International Conferences On the Move to Meaningful Internet Systems*, pages 1223–1237, 2002.

21 Konstantinos Skianis, Fragkiskos Malliaros, and Michalis Vazirgiannis. Fusing document, collection and label graph-based representations with word embeddings for text classification. In *12th Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs)*, New Orleans, Louisiana, USA, 2018.

22 Yangqiu Song, Shyam Upadhyay, Haoruo Peng, Stephen Mayhew, and Dan Roth. Toward any-language zero-shot topic classification of textual documents. *Artificial Intelligence*, 274:133–150, 2019.

23 R. Speer and Joshua Chin. An Ensemble Method to Produce High-Quality Word Embeddings. arXiv, 2016. `arXiv:1604.01692`.

24 Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In *31st AAAI Conference on Artificial Intelligence*, 2017.

25 Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. How to Fine-Tune BERT for Text Classification? arXiv, 2019. `arXiv:1905.05583`.

26 Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. Learning to rank answers on large online qa collections. In *46th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 719–727, 2008.

27 Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. arXiv, 2017. `arXiv:1706.03762`.

28 Felix Wu, Tianyi Zhang, Amauri Holanda de Souza Jr, Christopher Fifty, Tao Yu, and Kilian Q Weinberger. Simplifying graph convolutional networks. arXiv, 2019. `arXiv:1902.07153`.

29 Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5753–5763, 2019.

30 Wenpeng Yin, Jamaal Hay, and Dan Roth. Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach. arXiv, 2019. `arXiv:1909.00161`.

31 Jingqing Zhang, Piyawat Lertvittayakumjorn, and Yike Guo. Integrating semantic knowledge to tackle zero-shot text classification. arXiv, 2019. `arXiv:1903.12626`.

Relevance Feedback Search Based on Automatic Annotation and Classification of Texts

Rafael Leal
HELDIG Centre for Digital Humanities, University of Helsinki, Finland

Joonas Kesäniemi
Semantic Computing Research Group (SeCo), Aalto University, Finland

Mikko Koho
HELDIG Centre for Digital Humanities, University of Helsinki, Finland
Semantic Computing Research Group (SeCo), Aalto University, Finland

Eero Hyvönen
Semantic Computing Research Group (SeCo), Aalto University, Finland
HELDIG Centre for Digital Humanities, University of Helsinki, Finland

Abstract

The idea behind Relevance Feedback Search (RFBS) is to build search queries as an iterative and interactive process in which they are gradually refined based on the results of the previous search round. This can be helpful in situations where the end user cannot easily formulate their information needs at the outset as a well-focused query, or more generally as a way to filter and focus search results. This paper concerns (1) a framework that integrates keyword extraction and unsupervised classification into the RFBS paradigm and (2) the application of this framework to the legal domain as a use case. We focus on the Natural Language Processing (NLP) methods underlying the framework and application, where an automatic annotation tool is used for extracting document keywords as ontology concepts, which are then transformed into word embeddings to form vectorial representations of the texts. An unsupervised classification system that employs similar techniques is also used in order to classify the documents into broad thematic classes. This classification functionality is evaluated using two different datasets. As the use case, we describe an application perspective in the semantic portal *LawSampo – Finnish Legislation and Case Law on the Semantic Web*. This online demonstrator uses a dataset of 82 145 sections in 3725 statutes of Finnish legislation and another dataset that comprises 13 470 court decisions.

2012 ACM Subject Classification Computing methodologies → Information extraction; Applied computing → Document searching; Information systems → Clustering and classification

Keywords and phrases relevance feedback, keyword extraction, zero-shot text classification, word embeddings, LawSampo

Digital Object Identifier 10.4230/OASIcs.LDK.2021.18

1 Introduction

In many search situations, the information need of the user cannot be formulated precisely. The search query must then be gradually refined and the results re-evaluated in a series of successive rounds in order to achieve a satisfactory outcome. This paper describes language technology algorithms used in the application of this kind of iterative and interactive method, the Relevance Feedback Search (RFBS) paradigm [1, Ch. 5], to the search and exploration of textual documents. We outline a search system that integrates keyword extraction and unsupervised categorization of documents into RFBS based on pre-trained models and algorithms. We also present a case study with an implementation of this framework to the legal domain as part of the *LawSampo – Finnish Legislation and Case Law on the Semantic Web*[1] [6] system.

[1] Project homepage: `https://seco.cs.aalto.fi/projects/lakisampo/en/`

Algorithm 1 Proposed relevance feedback search.

Result: Result set RS in documents D satisfying the user's information need
/* Initialization */
1 AQ := ε ; // Active free-form text query (initially the empty string)
2 AK := $\emptyset$; // Active Keywords set (initially empty)
3 AC := $\emptyset$; // Active Categories set (initially empty)
4 RS := $\{d_0, ..., d_n\}$; // Result Set (initially contains the whole document domain)
/* RFBS loop */
5 **while** *RS is not a satisfying result* **do**
6 SK := KeywordsOf(RS) ; // Suggested keywords based on RS
7 SC := CategoriesOf(RS) ; // Suggested categories based on RS
8 AQ := ModifyQ(AQ) ; // User optionally modifies AQ
9 AK := ModifyK(AK, KW) ; // User optionally modifies AK based on SK
10 AC := ModifyC(AC, SC) ; // User optionally modifies AC based on SC
11 RS := Search(AQ, AK, AC);
12 **end**

The proposed RFBS process is outlined in Algorithm 1. The function $Search(AQ, AK, AC)$ in line 11 is used to search documents based on the search query AQ, active keywords AK, and active categories AC. The idea is that the categories are used for setting larger thematic contexts for the search, which can then be refined using keywords. For example, categories may refer to different phases or situations during the life of the end users, such as childhood or golden age [18], or societal contexts, such as health or environmental issues. In each iteration of the RFBS loop (lines 5–12), the system computes a set of new categories and keywords based on the search results RS as suggestions for the user to consider. The functions *ModifyQ*, *ModifyK* and *ModifyC* in lines 8-10 allow the user to make optional modifications for the next search round. In this way, the process is expected to converge gradually towards more and more satisfying results in RS.

A novel idea in the proposed approach is to combine implicit and explicit feedback methods [15] by using topical classification of the documents and keyword concepts extracted from the search results. User feedback on the topics and keywords is used to generate new search queries, thus guiding the iterative search process.

In order to avoid the challenges of traditional text-based search methods, for example morphological variation of words in highly inflectional languages such as Finnish, and semantic difficulties such as synonymy and polysemy, the presented system works on a semantic level. This is done based on ontological concepts and themes extracted automatically from the texts, which are processed via a word embedding algorithm. A zero-shot classification sub-system based on the same approach is responsible for categorizing the documents.

The implementation of the system uses Finnish legal documents, in particular sections of the law and court decisions, with the aim to help users to find jurisprudence related to their life situations. For the data, we use documents from the Semantic Finlex data service[2] [11], refined further for the LawSampo system and data service [6]. This search prototype constitutes one application perspective[3] of the LawSampo system. It is designed to work with Finnish language content, but the methods presented here can be applied to documents in any language, if similar ontologies and language processing tools are available.

[2] `https://data.finlex.fi/`
[3] In Sampo portals, the different ways for accessing the data are called "perspectives".

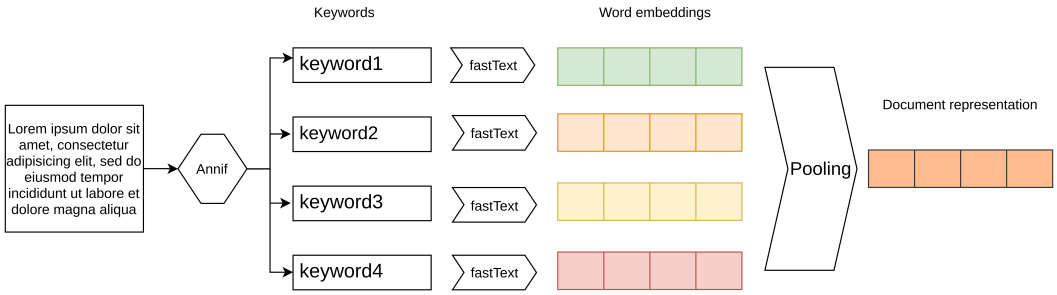


Figure 1 Simplified visual overview of the document representation building algorithm. The *document matrix* D collects the representations for all the documents in the dataset.

In the next sections, the methods and software needed for knowledge extraction of thematic categories and keywords from the texts are described, alongside some evaluation results. Subsequently, the data underlying our case study is explained, as well as the RFBS application in LawSampo. Finally, the contributions of our work are discussed alongside related works, and directions for further research are suggested.

2 Framework for Knowledge Extraction and Search

All documents in the target dataset are semantically annotated in two ways: 1) by extracting keyword concepts based on a keyword ontology and 2) by classifying the documents into a set of larger thematic categories. The search framework relies on three main components: the internal representation of the documents, the classifier, and the search algorithm. These will be explained in more detail in the following subsections.

2.1 Document Representation

The core of this system lies in its representations of the documents, which are built using a three-step approach: 1) representative keywords are extracted from the text, then 2) transformed into their respective word embeddings, and finally 3) combined via mean pooling to form the document representation vector. This process is visualized in Figure 1.

Keyword extraction is performed via Annif [4] [21], a subject indexing tool developed by the National Library of Finland. It is capable of using different algorithms to return suggested keywords and their respective weights for a given input text. Annif developers also provide various models, with different combinations of algorithms, on the Finto REST API[5]. They are all trained on bibliographical metadata from the works found on the Finna portal[6], which publishes information about objects in Finnish archives, museums and libraries, in a vein similar to Europeana. Since the training data is labelled with terms from the General Finnish Ontology (YSO)[7] [17] and its sister ontology YSO places (YSO-paikat), the API returns keywords identified by unique YSO URIs. This also means that the keywords obtained are already lemmatized. YSO contains a total of 31 205 main concept labels and an additional 7807 are available in YSO places.

[4] https://annif.org/
[5] https://ai.finto.fi/
[6] https://www.finna.fi/
[7] https://finto.fi/yso/en/

Our present system uses the *yso-fi* pre-trained model on the Finto REST API. This model is an ensemble, with equal weights, of three algorithms:

- a modified version of **TF-IDF**, which pre-calculates TF-IDF vectors for each *keyword* and compares them to similarly built document vectors to find the best matches [21, p. 8];
- **Maui**, an n-gram-based keyword extractor [9];
- **Parabel**, an extreme multi-label classification algorithm [13].

The first two directly match salient terms to a vocabulary, while the last is also able to find indirect correlations between words [21]. This mix provides results that are not only grounded on the text of the documents but also able to extrapolate their specific wording. A limit of 100 keywords per text and a minimum weight threshold of 0.01 are used, which offer a wider range of results than the defaults (respectively 10 and 0) and proved effective in the classification task [8]. The keyword weights obtained from Annif are saved into the *keyword matrix* K, a document-term structure.

Once the texts are distilled into sets of representative keywords, our system uses fastText [3] to obtain word embeddings for each of them. This combination of algorithms improves the task of unsupervised categorization by avoiding creating representations at the sentence level. In [8], various combinations of text representation and embedding algorithms were tested, including transformer-based algorithms such as FinBERT and S-BERT/XLM-R. Nevertheless, it concluded that fastText, especially in partnership with Annif, produced the best results using this technique.

FastText improves on the word2vec embedding algorithm by breaking up the words into a bag of n-grams, each with their own vectorial representation. When building the final embedding, these components are also taken into account. This mechanism provides the means to build representations for words that are not in the training data by breaking them up into smaller, already-seen chunks. This is especially important for morphologically rich languages such as Finnish, which present unseen word forms more often than analytic languages.

Representations are calculated via the `get_sentence_vector` function from the fastText Python package, which averages the l2 norm of the respective representation vector for each word in the sentence. The system uses a pre-trained language model offered by the fastText developers, trained on the Wikipedia (2017 dump) and Common Crawl datasets [5]. It has a dimension of 300 and a vocabulary containing the 2 million most common word tokens in the training data – around 18K concept labels from YSO are nevertheless not among them.

The final representation for each document is obtained via mean pooling of their keyword embeddings. All document representations are then collected into a $D^{d\times 300}$ *document matrix*, where d is the total number of documents and 300 is the number of dimensions in the pre-trained fastText model.

2.2 Document Classification

A similar process is used to build a $C^{c\times 300}$ *category matrix*, where c is the total number of category labels. Although the keyword extraction algorithms are trained on categorical metadata, as explained in Section 2.1, the classification mechanism that employs it can be considered unsupervised, since it is built without any training.

Category labels must be provided by the end user, since at this point the system does not recognise broad topics automatically. The labels are pre-processed by stripping commas and conjunctions and used as input for the category label representation pipeline, illustrated in Figure 2.

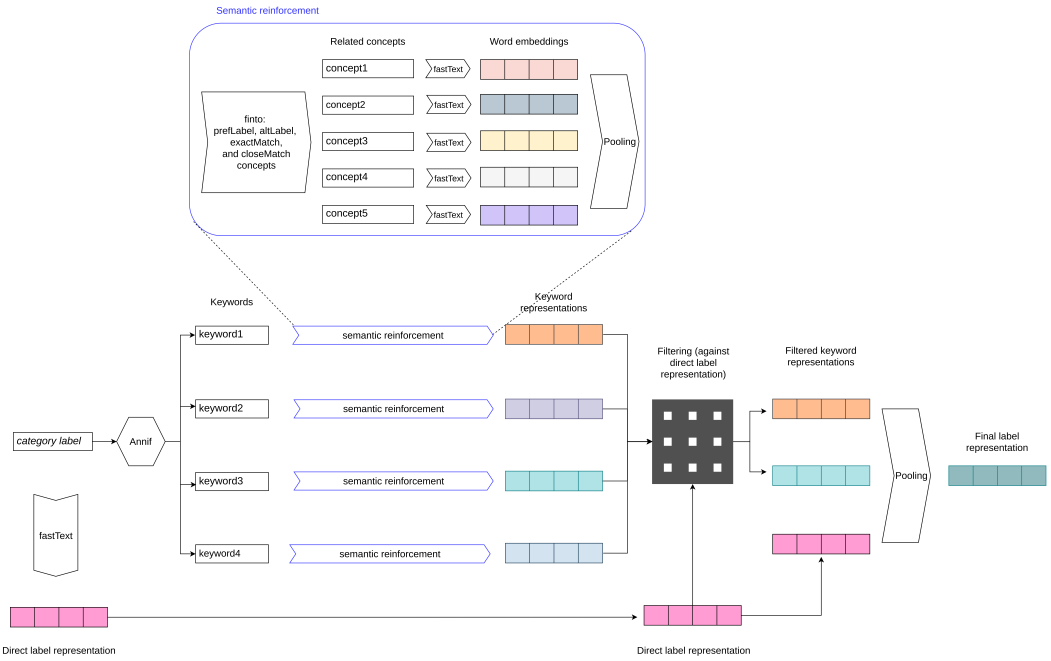


Figure 2 Simplified visual overview of how category label representations are built. First, related keywords obtained via Annif are semantically reinforced and transformed into vectors, then they are compared to an embedding of the original label (*direct label representation*). Those which score above a threshold are pooled together into the *final label representation*. The *category matrix C* contains the final label representations for all category labels.

Annif's results can be somewhat erratic when it comes to very short texts, and the category labels often contain a single word. Thus, filtering out unrelated terms from the set of keyword suggestions can thus produce better results. This is done via cosine similarity: a matrix containing word embeddings for the entire category label text as well as its individual words is built, and then compared to another matrix containing *semantically reinforced* embeddings for all the keywords suggested by Annif (this reinforcement technique is described below). If the maximum cosine similarity between a suggestion and any of the query vectors is under a certain threshold (0.25 is the default), the keyword is rejected. The word embeddings for the original label and the resulting keywords suggestions, pooled together, form the category representation vector.

Since Annif returns YSO concepts, their vectorial representation can be semantically reinforced. This simple technique is based on ontology relations: main and alternative labels (*prefLabel* and *altLabel*), exactly matching (*exactMatch*) and closely matching (*closeMatch*) entities linked to the target concept are fetched; the final representation is calculated as the average embedding of this expanded set of entities.

As an example, the concept *shares*, whose URI is `http://www.yso.fi/onto/koko/p12994`, returns a wealth of associated concepts: broader, narrower, related, exactly matching and closely matching concepts, as well as sections named "in other languages" and "entry terms", which are alternative labels for the concept. Out of those, the framework chooses only the following:

- ***prefLabel*** (the main label): *shares*
- ***altLabel*** (entry terms): *share, stocks*
- ***closeMatch*** (closely matching categories): *Stocks* (linked to the Library of Congress concept)

- ***exactMatch*** (exactly matching categories): *share, shares* (linked to other Finnish ontologies)

The associated terms are then collected in a set $S = \{shares, share, stocks\}$ and the final representation will be an average of the word embeddings for the terms in S. In this specific case, the procedure helps disambiguate this concept from another *shares* entry[8], which represents technical objects also known as *drill coulters* or *ploughshares*. However, this technique can help build better representations even without disambiguation, by pooling together similar concepts instead of relying on a single label.

The dot product between the *category matrix* C and the *document matrix* D builds a $V^{c \times d}$ *category-document matrix*, which stores the strength of each category-document pair.

2.3 Search

The search results obtained by our RFBS framework depends on four elements: text query, category, positive keywords, and negative keywords. They may work separately or in tandem. This algorithm corresponds to the *Search*() method in line 11 of Algorithm 1, in which both positive and negative keyword sets are represented by AK.

The free-form search query, in case it is given, is the first element to be resolved. It is transformed into embeddings in the standard way, via Annif→fastText. The results can be filtered as explained in section 2.2, however without semantic reinforcement in order to save computational time. The most similar documents are calculated via cosine similarity between the query representation and the *document matrix* D, and the results are collected in a result list.

Positive keywords and categories are executed next, in case they are set. First, scores for each document are calculated as the sum of their respective weights (from the keyword matrix K) for all the positive keywords in question. These scores are used either as weights for an already-existing result list or to create a result list from scratch. Category scores are simply the respective row of the *category-document matrix* V. Similarly to the previous step, these scores are used either as weights or to create a new result list in case no other search parameters are set. Finally, negative keywords are used to exclude all documents from the result list containing any of the negative keywords chosen.

Next, the system calculates keywords candidates and category candidates. These are equivalent to the methods *KeywordsOf*() and *CategoriesOf*() in lines 6 and 7 of Algorithm 1. Keywords are calculated by averaging the respective *keyword matrix* K scores for all documents in the result list. Additionally, category candidates are calculated by averaging the category scores for the relevant documents in the *category-document matrix* V.

3 Evaluation of the System: Current Status

In this section, we present an overview of the evaluation of the framework components, focusing on the classification system.

3.1 Classification

The classification component of this system has been evaluated with two different datasets. Neither is representative of the dataset in the Use Case, so they can be both considered baselines. The datasets are the following:

[8] `http://www.yso.fi/onto/koko/p48662`

- A custom Yle dataset, containing 5096 news articles from Yle (Yleisradio Oy), the Finnish Broadcasting Company[9], classified in 11 categories according to their main tag: *politiikka*, 'politics', *talous* 'economy', *kulttuuri* 'culture', *luonto* 'nature', *tiede* 'science', *terveys* 'health', *liikenne* 'transportation', *urheilu* 'sports', *sää* 'weather', *parisuhde* 'intimate relationships' and *rikokset* 'crimes'). This dataset is equivalent to *Yle 1* and *Yle 2* in [8] combined[10]. It has a mean length, in characters, of 2110.8 ± 1534.5.
- The Minilex dataset: a collection of 2567 legislation-related questions classified in 13 categories (*asunto, kiinteistö* 'housing, real estate', *immateriaalioikeus* 'intellectual property', *irtain omaisuus* 'chattel', *lainat, velat* 'loans, debts', *liikenne* 'transportation', *oikeudenkäynti* 'legal proceedings', *perheoikeus, perintöoikeus* 'family law, inheritance law', *rikokset* 'crimes', *sopimus, vahingonkorvaus* 'contracts, compensation', *työsuhde, virkasuhde* 'employment, public service', *ulosotto, konkurssi* 'debt recovery, bankruptcy', *vuokra* 'rent', *yritykset, yhteisöt* 'companies, organisations') from the legal services website Minilex[11]. As can be observed, there is semantic overlap among some of the categories in this dataset, such as "debt recovery, bankruptcy" / "loans, debts"; or "housing, real estate" / "rent" / "contracts, compensation"; or even "transportation" / "crimes". The mean length of his dataset is 447.3 ± 236.1 characters.

Three different measures were taken: the F_1 score, the Mean Reciprocal Rank (MRR), and the rank of the gold label among the predictions made by the classifier. The last metric springs from the fact that this system is used as a multi-label classifier, so not having the gold tag as the best prediction is not necessarily a consequential result.

The system fares better with the Yle dataset, with an F_1 score of 0.737, an MRR of 0.828 and a mean rank of 0.7 ('0' being the gold label, '1' the second prediction, etc; lower is thus better), while the Minilex dataset obtains an F_1 score of 0.56, an MRR of 0.697 and a mean rank of 1.557. Using semantically reinforced vectors gives a small boost to these numbers: Yle gets an F_1 of 0.742, an MRR of 0.832 and a mean rank of 0.67, and Minilex respectively 0.572, 0.705 and 1.496. Table 1 details the counts and cumulative distributions of the results.

These numbers show that the present classification method is capable of categorizing the gold labels as the top prediction in 57–74% of the cases and among the top 3 predictions in 80–90% of the cases, depending on the dataset used. This variation can possibly be attributed to a combination of the length of the documents and the quality of the category labels. This classifier tends to fare better with longer texts [8], so the Yle dataset, which contains texts that are around four times longer on average, has a starting advantage. Moreover, as discussed above, the Minilex labels are semantically more similar, which increases the probability of obtaining non-optimal predictions.

Legislative texts may present more difficulties for this system, since they contain a more peculiar and jargonic choice of words. Lack of familiarity with the words is a challenge for both the keyword-extracting and the word embedding algorithms, which may not recognize or represent them properly.

At any rate, the results presented here are not in line with state-of-the art supervised algorithms, which reach results above 95% and nearing 100% for their top predictions (cf. survey in [10]). However, training data in Finnish is hard to obtain, and without it a

[9] `https://yle.fi`
[10] Excluding one article, whose main tag has changed. This data is not redistributable.
[11] `https://www.minilex.fi`. Their terms of use allows for non-commercial usage of the data, but not for its redistribution.

Table 1 Count and cumulative distribution (CUMD) of results in the classification task for the Yle and Minilex datasets, in both their standard and reinforced versions.

	Yle				Minilex			
	Standard		Reinforced		Standard		Reinforced	
rank	COUNT	CUMD	COUNT	CUMD	COUNT	CUMD	COUNT	CUMD
0	3757	0.737	3779	0.742	1438	0.56	1469	0.572
1	560	0.847	568	0.853	406	0.718	388	0.723
2	269	0.9	264	0.905	179	0.788	187	0.796
3	147	0.929	151	0.934	102	0.828	104	0.837
4	115	0.951	96	0.953	90	0.863	78	0.867
5	91	0.969	104	0.974	59	0.886	64	0.892
6	68	0.983	57	0.985	86	0.919	86	0.926
7	38	0.99	30	0.991	61	0.943	62	0.95
8	25	0.995	22	0.995	56	0.965	44	0.967
9	19	0.999	18	0.999	39	0.98	38	0.982
10	7	1.0	7	1.0	32	0.993	28	0.993
11	-	-	-	-	18	1.0	18	1.0
12	-	-	-	-	1	1.0	1	1.0

supervised system cannot be built. The main advantages of the classification algorithm presented in this paper are its flexible nature – since it only requires a list of category labels in order to work – and its straightforward integration into the search framework, since it uses the same underlying technologies.

3.2 Document Representation

These results also show strong consistency on the part of Annif: it is capable of coherently assigning keywords to both documents and category labels, so that most documents can be correctly classified when transposed to a vector space of embeddings. FastText also demonstrates reliability performing this transposition from semantic meanings to vectorial representations.

A more decisive evaluation of this component is planned as part of our future work.

3.3 Search

No formal evaluation of the search system has been carried out so far. However, Section 4 contains tests and insights about the reliability and usability of this component.

4 Use Case: Search Engine for Finnish Legislation and Case Law

This section describes, via a concrete use case, how the RFBS has been adapted to the legal domain as part of the semantic portal LawSampo [6]. LawSampo is the first Sampo portal to add RFBS-based functionality to complement the search features of previous Sampos, which are mostly facet-based.

4.1 The Data

LawSampo contains data about Finnish legislation and case law as a harmonized Linked Data knowledge graph. This knowledge graph is based on Semantic Finlex data [11], filtered and transformed into a simpler data model. This was done with the aim of hiding the inherent complexity of legal documentation while keeping the data relevant to anyone interested in the topic. These data transformations were implemented as SPARQL CONSTRUCT queries, and the data is enriched with information about referenced EU legislation, as well as the generated annotations of subject keywords and category labels.

The set of category labels used in LawSampo's RFBS system is based on the Minilex dataset discussed in Section 3. However, in order to avoid the semantic overlap present among some of the categories and to expand the field of possible categories, they were refined with some input from experts at the Ministry of Justice of Finland. The resulting set contains 12 categories: *asuminen, kiinteistö* 'housing, real estate', *ihmisoikeudet, perusoikeudet* 'human rights, basic rights', *omaisuus, kaupankäynti, kuluttajansuoja* 'property, commerce, consumer protection', *julkishallinto, valtionhallinto* 'public administration', *rahoitus* 'finance', *verotus* 'taxation', *yritykset, yhteisöt, työelämä* 'business, organizations, working life', *liikenne, kuljetus* 'traffic', *perheoikeus, perintöoikeus* 'family law, inheritance', *rikosasiat, oikeudenkäynti* 'crime, legal proceedings', *koulutus* 'education', *ympäristö* 'environment'.

The consolidated legislation consists of 3725 statutes and their 82 145 sections, with each section consisting of the most current version of the full-text contents in Finnish. The case law dataset consists of 13 470 court decisions and their full-text contents.

4.2 LawSampo User Interface for RFBS

The LawSampo portal implements the RFBS Algorithm 1 in its "Contextual Search" application perspective[12]. LawSampo allows the user to initialize the RFBS system either by setting a free-form text query or by selecting one of the provided document categories from the list presented in Section 4.1. Figure 3 illustrates the user interface after the user has started a search via a text query. After the initial search, the application switches to an iterative mode corresponding to the while-do loop (lines 5-12) of Algorithm 1, where the search is fine-tuned by managing a set of active filters through categories and keywords suggested by the system. Since no category was selected in the initial phase in Figure 3, the user is presented with a top-three list of suggested categories (obtained in line 7 in the algorithm) in addition to 20 suggested keywords (line 6). In this implementation, the values used for the initial search cannot be changed during the iterative loop: if the initial filter (query or category) is removed, the iteration stops and the search returns to its empty initial state.

Table 2 shows a simple example of how, using the same query, "right of redemption", different categories affect the resulting documents. With the given query, it is not surprising that *Act on the Redemption of Immovable Property and Special Rights* is the most relevant hit. More interestingly, the *Water Act* becomes the second-most referenced document when the "human rights, basic rights" category is active, whereas choosing the "crime, legal proceedings" category surfaces the *Building Act*. Finally, the "property, commerce and consumer protection" category adds *Act on the Residential and Commercial Property Information System* to the returned documents.

[12] In the Sampo model, the user is provided with several independent but interlinked applications that use a shared underlying knowledge graph.

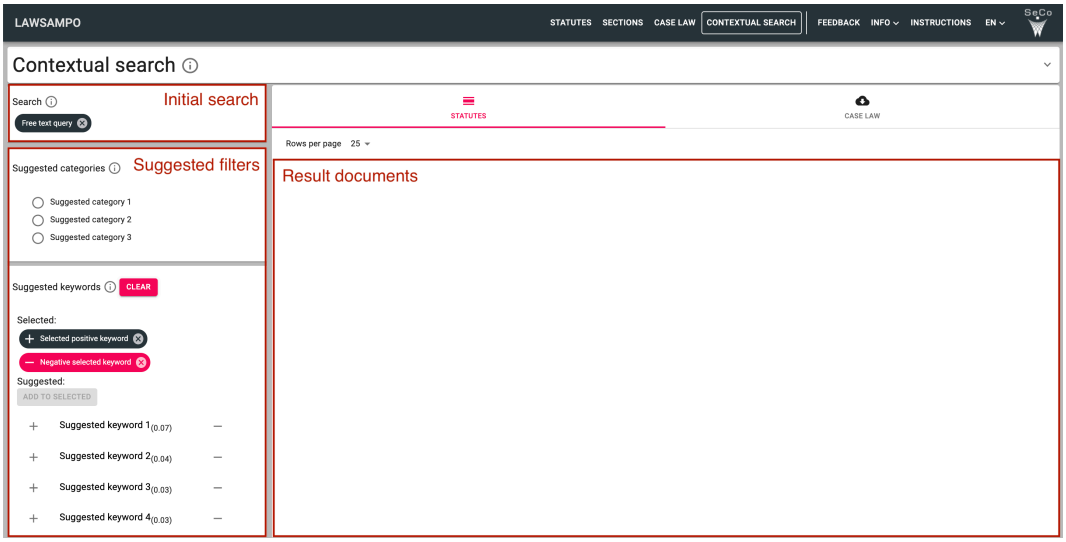


Figure 3 LawSampo's contextual search UI. The user has made an initial text query and selected one positive and one negative keyword. Next, the user can continue the search by using the suggested categories and keywords in order to modify the active filters. The number shown in subscript next to the keywords is a normalized relevance score.

Table 2 How does the selected category affect the resulting documents? This table shows statute results for the query *lunastusoikeus* 'right of redemption' with three different categories and a result size of five. The values represent the number of sections returned from each statute.

	crime, legal proceedings	**human rights, basic rights**	**property, commerce, consumer protection**
Water act	1	2	1
Building act	1		
Real Estate Formation Act	1		1
Act on the Residential and Commercial Property Information System			1
Act on the Redemption of Immoveable Property and Special Rights	2	3	2

Suggested keywords are shown with a relevance score calculated on the basis of the current list of resulting documents. Keywords can be added to the active filters (line 9 in Algorithm 1) in either positive or negative mode using the plus (+) or minus (−) buttons respectively. A negative selection excludes any documents containing the given keyword from the result list, as explained in Section 2.3.

The result list view can be toggled between statutes and case law documents as two parallel tabs. The results are updated whenever the user changes any of the active filters, i.e., textual query, category or selected positive and negative keywords. The documents returned are statute sections or case law abstracts. The user is given the possibility to skim through them and to follow links to the full documents. Since the statutes search works at the section level, the results can contain multiple documents from the same statute.

Table 3 Relevance feedback development during the first search iterations in the example scenario. *Iteration 0* is the initial search, which is done without any selected keyword filters. Positive and negative keywords are denoted with "+" and "−" characters respectively. The *Suggested keywords* row only shows a subset of the keywords suggested by LawSampo's contextual search. The italic typeface is used to mark keywords selected for the next iteration.

	Iteration 0	**Iteration 1**	**Iteration 2**	**Iteration 3**
Query	"expropriation lot"			
Selected keywords		+ redemption + right for redemption − alluvial land − compulsory auction	+ redemption + right for redemption − alluvial land − compulsory auction	+ compensation for redemption − alluvial land − compulsory auction
Document type	Statutes	Statutes	Case law	Statutes
Suggested keywords	real property, *redemption*, *right for redemption* cadastral procedures, land surveying, *alluvial land*, *compulsory auction*	savings banks, railways, limited companies, shares, town planning, land use policy, company law	roads, construction, municipalities, land use planning, land acquisition, land use, *compensation for redemption*	indemnities, prices, value (properties), interest (economics), owners

4.3 Search Example Scenario

This section presents as an example of how, using LawSampo's "Contextual Search", the following information need can be satisfied:

> *I've been thinking about making a huge renovation to our house. However, I'm worried that because our house is in such a good area, the city might expropriate the lot. If that happens, what kind of payout could I be looking at?*

Let us begin the search with a simple query based on the information need described above: *pakkolunastus tontti* 'expropriation lot'. The query is executed against the statutes by default and, for this example, the maximum number of results is set to 10. A summary of the RFBS process for each search iteration can be found in Table 3.

The document list resulting from the first iteration (RS_1) does not look very promising: the only vaguely relevant document is *Erämaalaki* 'Wilderness Act', which describes how the state can expropriate land in wilderness areas in order to build roads. The result list also contains multiple non-relevant documents related to forced auction and water systems. For the next iteration, we add from the suggested keywords *lunastus* 'redemption' and *lunastusoikeus* 'right of redemption' as positive keywords and *vesijättömaa* 'alluvial land' and *pakkohuutokauppa* 'compulsory auction' as negative ones to our set of active filters (AK in Algorithm 1).

The second set of results (RS_2) is already more useful. There are two more references to expropriation related to roads and railroads that can be considered somewhat relevant, but also a match to the highly useful *Land use and Building Act*. The results also indicate that

the positive keyword filters might not work as intended, since the results include documents related to *limited companies*, which deal with the wrong kind of "redemption" with respect to our information need. The results even contain a section from the *Saving Back Act*, most likely due to the use of the term *redemption* in the document.

We can test our intuition about the keyword "redemption" by switching over to the case law view to verify if the results are similar: the case law results (RS_3) with the same set of active filters are indeed consistent with the results in the statute view, with a couple of only indirectly relevant documents. However, the suggested keywords (AK_3) now include *lunastuskorvaukset* 'compensation for redemption', which matches our information need perfectly.

Finally, let us replace the keywords "redemption" and "right of redemption" with "compensation from redemption" and swap back to the statutes view to retrieve one more result list (RS_4). This time, all returned documents can be considered useful, including three references to a document titled *Act on the Redemption of Immovable Property and Special Rights*.

5 Discussion

This section overviews earlier related research, summarizes the contributions made by this paper, and outlines paths for future research.

5.1 Related Work

Various methods exist for relevance feedback search [1, 15]. Teevan et al. [23] enrich web search with relevance feedback based on a constructed user profile. Peltonen et al. [12] combine visual intent modeling with exploratory relevance feedback search. Tang et al. [22] used topic modeling in academic literature search. Song et al. [20] employed topic modeling with relevance search, based on implicit feedback from the topics of the user web search history. In [7], RFBS is combined with topic modelling [2]. However, the method of combining automatic document classification and keyword extractions with relevance feedback search, as described in this paper, is novel.

Regarding zero-shot classification methods, most of them work by training a classifier and then adapting it to a new set of categories [4, 14, 24], while [26] also integrates a knowledge graph into their algorithm. Among unsupervised classification models, [16] use skip-gram word embeddings to calculate the semantic similarity between a label and the given documents, which is also the basis of our work. In contrast, [25] treats zero-shot as an entailment problem.

5.2 Contributions

This paper argued for using a combination of topical classification and ontological keywords as a semantic basis for RFBS when exploring textual documents from complex domains, such as legislation and case law. A method for accomplishing this was presented, as well as an implementation of it for testing and evaluating purposes.

The content annotation results shown in Section 3 indicate that the proposed classification system, despite its unsupervised nature, is capable of classifying documents correctly 74% of the time (or 90% within the first three predictions) when the classes are semantically non-overlapping and the texts are long enough.

Section 4 illustrated how the RFBS algorithm suggested in this paper can be used in practice, demonstrating how LawSampo's Contextual Searcher perspective can be used to navigate documents from a semantically complex domain successfully in an iterative fashion.

Automatically suggested keywords mitigate the burden on the user of coming up with suitable queries and can provide valuable feedback even when the user selects non-optimal keywords. We have not yet performed a more general evaluation of the search functionality besides testing the system in selected individual problems. Nevertheless, the experiments presented in this paper suggest that a combination of ontological keyword annotations and topical classifications with word embeddings can create a useful semantic basis for the RFBS paradigm when searching and exploring textual legal documents.

5.3 Future Work

More research will be done in order to improve the vectorial representation of the documents. As it stands, these representations are entirely based on each document's set of keywords, which add depth (by emphasizing these keywords) but subtract breadth (other details of the text) from them. We plan to pursue three main lines of research in the future: one line aims at improving the set of representative keywords by both filtering out unrelated suggestions and adding new ones via ontology relations, named entity recognition and other keyword extraction algorithms, provided they can be integrated into the system. The second line aims at investigating alternative representation vectors partly based on whole-text embeddings. A third line of research consists in the automatic identification of major topics in the data as an alternative to the user-provided category list: this can possibly be accomplished by capitalizing on existing clustering methods and ontological relations among the keywords.

LawSampo is the first portal in the Sampo series of systems[13] to take advantage of this search functionality based on relevance feedback. Similar methods are planned as part of the upcoming new Sampos as well. These include especially the ParliamentSampo system[14], which incorporates over 900 000 parliamentary debate speeches [19] from the Parliament of Finland (1907–2021), documents that are related to the legislative texts found in LawSampo.

References

1 R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval (2nd Ed.)*. Addison-Wesley Longman Publishing Co., Inc., 2011.

2 David M. Blei. Probabilistic topic models. *Commun. ACM*, 55(4):77–84, 2012. doi:10.1145/2133806.2133826.

3 Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5:135–146, December 2017. doi:10.1162/tacl_a_00051.

4 Yann N. Dauphin, Gokhan Tur, Dilek Hakkani-Tur, and Larry Heck. Zero-Shot Learning for Semantic Utterance Classification. *ICLR 2014*, 2014. arXiv:1401.0509.

5 Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

6 Eero Hyvönen, Minna Tamper, Arttu Oksanen, Esko Ikkala, Sami Sarsa, Jouni Tuominen, and Aki Hietanen. LawSampo: A semantic portal on a linked open data service for finnish legislation and case law. In *The Semantic Web: ESWC 2020 Satellite Events. Revised Selected Papers*, pages 110–114. Springer–Verlag, 2019.

7 Mikko Koho, Erkki Heino, Arttu Oksanen, and Eero Hyvönen. Toffee - semantic media search using topic modeling and relevance feedback. In *Proceedings of the ISWC 2018 Posters &*

[13] https://seco.cs.aalto.fi/applications/sampo/
[14] https://seco.cs.aalto.fi/projects/semparl/en/

Demonstrations, Industry and Blue Sky Ideas Tracks. CEUR Workshop Proceedings, October 2018. Vol 2180. URL: http://ceur-ws.org/Vol-2180/.

8 Rafael Leal. Unsupervised zero-shot classification of Finnish documents using pre-trained language models. Master's thesis, University of Helsinki, Department of Digital Humanities, 2020. URL: http://urn.fi/URN:NBN:fi:hulib-202012155147.

9 Olena Medelyan. *Human-Competitive Automatic Topic Indexing*. Thesis, The University of Waikato, 2009. URL: https://researchcommons.waikato.ac.nz/handle/10289/3513.

10 Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. Deep learning–based text classification: A comprehensive review. *ACM Computing Surveys (CSUR)*, 54(3):1–40, 2021. doi:10.1145/3439726.

11 Arttu Oksanen, Jouni Tuominen, Eetu Mäkelä, Minna Tamper, Aki Hietanen, and Eero Hyvönen. Semantic Finlex: Transforming, publishing, and using finnish legislation and case law as linked open data on the web. In G. Peruginelli and S. Faro, editors, *Knowledge of the Law in the Big Data Age*, volume 317 of *Frontiers in Artificial Intelligence and Applications*, pages 212–228. IOS Press, 2019. ISBN 978-1-61499-984-3 (print); ISBN 978-1-61499-985-0 (online). URL: http://doi.org/10.3233/FAIA190023.

12 Jaakko Peltonen, Jonathan Strahl, and Patrik Floréen. Negative relevance feedback for exploratory search with visual interactive intent modeling. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, pages 149–159. ACM, 2017.

13 Yashoteja Prabhu, Anil Kag, Shrutendra Harsola, Rahul Agrawal, and Manik Varma. Parabel: Partitioned Label Trees for Extreme Classification with Application to Dynamic Search Advertising. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web - WWW '18*, pages 993–1002. ACM Press, 2018. doi:10.1145/3178876.3185998.

14 Anthony Rios and Ramakanth Kavuluru. Few-Shot and Zero-Shot Multi-Label Learning for Structured Label Spaces. *EMNLP*, 2018. doi:10.18653/v1/D18-1352.

15 Gerard Salton and Chris Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4):288, 1990.

16 Prateek Veeranna Sappadla, Jinseok Nam, Eneldo Loza Mencía, and Johannes Fürnkranz. Using semantic similarity for multi-label zero-shot classification of text documents. In *ESANN*, 2016.

17 Katri Seppälä and Eero Hyvönen. Asiasanaston muuttaminen ontologiaksi. Yleinen suomalainen ontologia esimerkkinä FinnONTO-hankkeen mallista (Changing a keyword thesaurus into an ontology. General Finnish Ontology as an example of the FinnONTO model). Technical report, National Library, Plans, Reports, Guides, March 2014. URL: https://www.doria.fi/handle/10024/96825.

18 Teemu Sidoroff and Eero Hyvönen. Semantic e-goverment portals - a case study. In *Proceedings of the ISWC-2005 Workshop Semantic Web Case Studies and Best Practices for eBusiness SWCASE05*, 2005. URL: https://seco.cs.aalto.fi/publications/2005/sidoroff-hyvonen-semantic-e-government-2005.pdf.

19 Laura Sinikallio, Senka Drobac, Minna Tamper, Rafael Leal, Mikko Koho, Jouni Tuominen, Matti La Mela, and Eero Hyvönen. Plenary debates of the Parliament of Finland as linked open data and in Parla-CLARIN markup, March 2021. Accepted, LDK 2021.

20 Wei Song, Yu Zhang, Ting Liu, and Sheng Li. Bridging topic modeling and personalized search. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1167–1175. Association for Computational Linguistics, 2010.

21 Osma Suominen. Annif: DIY automated subject indexing using multiple algorithms. *LIBER Quarterly*, 29(1):1–25, July 2019. doi:10.18352/lq.10285.

22 Jie Tang, Ruoming Jin, and Jing Zhang. A topic modeling approach and its integration into the random walk framework for academic search. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 1055–1060. IEEE, 2008.

23 Jaime Teevan, Susan T. Dumais, and Eric Horvitz. Personalizing search via automated analysis of interests and activities. In *Proc. of the 28th Annual International ACM SIGIR Conference*, SIGIR '05, pages 449–456. ACM, 2005.

24 Zhiquan Ye, Yuxia Geng, Jiaoyan Chen, Jingmin Chen, Xiaoxiao Xu, SuHang Zheng, Feng Wang, Jun Zhang, and Huajun Chen. Zero-shot Text Classification via Reinforced Self-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3014–3024. Association for Computational Linguistics, 2020. doi:10.18653/v1/2020.acl-main.272.

25 Wenpeng Yin, Jamaal Hay, and Dan Roth. Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3914–3923. Association for Computational Linguistics, 2019. doi:10.18653/v1/D19-1404.

26 Jingqing Zhang, Piyawat Lertvittayakumjorn, and Yike Guo. Integrating Semantic Knowledge to Tackle Zero-shot Text Classification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1031–1040. Association for Computational Linguistics, 2019. doi:10.18653/v1/N19-1108.

Automatic Construction of Knowledge Graphs from Text and Structured Data: A Preliminary Literature Review

Maraim Masoud
Insight SFI Research Centre for Data Analytics, Data Science Institute, National University of Ireland Galway, Ireland

Bianca Pereira
Insight SFI Research Centre for Data Analytics, Data Science Institute, National University of Ireland Galway, Ireland

John McCrae
Insight SFI Research Centre for Data Analytics, Data Science Institute, National University of Ireland Galway, Ireland

Paul Buitelaar
Insight SFI Research Centre for Data Analytics, Data Science Institute, National University of Ireland Galway, Ireland

Abstract

Knowledge graphs have been shown to be an important data structure for many applications, including chatbot development, data integration, and semantic search. In the enterprise domain, such graphs need to be constructed based on both structured (e.g. databases) and unstructured (e.g. textual) internal data sources; preferentially using automatic approaches due to the costs associated with manual construction of knowledge graphs. However, despite the growing body of research that leverages both structured and textual data sources in the context of automatic knowledge graph construction, the research community has centered on either one type of source or the other. In this paper, we conduct a preliminary literature review to investigate approaches that can be used for the integration of textual and structured data sources in the process of automatic knowledge graph construction. We highlight the solutions currently available for use within enterprises and point areas that would benefit from further research.

2012 ACM Subject Classification Information systems → Information extraction

Keywords and phrases Knowledge Graph Construction, Enterprise Knowledge Graph

Digital Object Identifier 10.4230/OASIcs.LDK.2021.19

Funding This work was supported by Science Foundation Ireland under grant SFI/12/RC/2289_P2 (Insight).

1 Introduction

The automatic construction of knowledge graphs from textual or structured data sources enables the generation of domain-specific enterprise knowledge graphs while decreasing the costs associated with manual generation of formal knowledge datasets [29]. The extraction from textual sources enables the representation of internal knowledge generated by employees and customers through the analysis of a domain of discourse, whereas the extraction from structured data sources (e.g. relational databases) enables the representation of enterprise information that is generated and applied in the provision of services and applications. Despite the potential benefits of both text and structured sources in the context of domain-specific enterprise knowledge graphs, the research community has focused on either one source of data or the other. In this paper, we perform a literature review that explores the

combination of both types of data sources in the process of constructing domain-specific enterprise knowledge graph. As such construction tends to be automatically performed, the literature review investigates how both data sources can be integrated in the process of automatic knowledge graph construction.

First we highlight the initiatives promoting knowledge graph construction either from text or structured data sources in Section 2. Then, in Section 3, we present our methodology for analysis of the literature. In Section 4 we introduce the conceptual framework used to analyse the literature for domain-specific knowledge graph construction. Further, we present the results in Sections 5. We conclude the paper by highlighting the solutions available, lessons learned from this analysis, and pointing directions for future research.

2 Automatic Construction of Knowledge Graphs

Automatic knowledge graph construction has been the target of research challenges and initiatives in different research communities [2, 23, 31, 11].

Initiatives such as SemEval and OAEI focus only on the use of either textual or structured data sources. The SemEval Taxonomy Extraction Task [2], organised as part of SIGLEX/SIGSEM[1] conference in 2016, proposed the extraction of domain-specific taxonomic structures exclusively from textual data sources. Whereas the OAEI [11] organised as part of the International Semantic Web Conference (ISWC) since 2006, focuses only on the use of structured data sources where the goal is to create an integrated ontology based on the alignment between two (or more) already available ontologies or knowledge graphs.

In the intersection between text and structured data, we can find the TAC-KBP and NEEL challenges that aim at verifying if entities appearing in text are already represented in a structured knowledge base. The TAC-KBP Entity Linking task [23], a yearly event from 2009 to 2016 as part of TAC[2], aimed at taking advantage of a preexistent general domain structured data source and use the text as source of additional information to expand it. In a similar manner, the NEEL challenge [31], that run from 2013 to 2016 as part of the WWW Conference, focused on identifying if entities appearing in microblog messages (i.e. tweets) were already available in a general domain knowledge base.

Despite all the available initiatives in the automatic construction of knowledge graphs based on different sources, none of these initiatives explicitly focuses on both: (i) the aggregated use of text and structured data sources, and (ii) the generation of a domain-specific knowledge graph. Therefore, the goal of this paper is to analyse what are the available approaches in the literature that could be applied to the automatic construction of domain-specific enterprise knowledge graphs based on textual and structured data sources and what are the areas the require further research.

3 Methodology

The methodology used to select the literature followed four steps (Figure 1): (i) selection of seed papers, (ii) search, (iii) filtering, and (iv) analysis.

The selection of seed papers was performed based on a convenience sample, with a selection of survey and literature review papers known to the authors. A set of six seed papers was selected [5, 25, 26, 32, 33, 45].

[1] The Special Interest Group on the Lexicon (SIGLEX)/Special Interest Group on Computational Semantics (SIGSEM)

[2] Text Analysis Conference

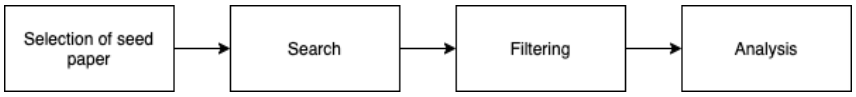


Figure 1 Methodology for literature review.

Table 1 Inclusion and exclusion criteria for selection of the literature.

Inclusion Criteria	Exclusion Criteria
▪ √written in the English language. ▪ √published in conferences proceedings or in a journal. ▪ √the abstract and conclusion indicate the paper is in the topic of enriching the results of automatic extraction of knowledge graph with structured data.	▪ ✕only (semi-)manual approaches. ▪ ✕no explanation of the method used. ▪ ✕using only textual data sources.

Based on these seed papers, we expand our search to include also the literature in their list of references. Next, all the literature collected is filtered according to a set of inclusion and exclusion criteria (Table 1). Finally, all papers that passed the filtering criteria are analysed according to a series of dimensions of interest.

4 Dimensions Used for Analysis of the Literature

The analysis of the literature was performed based on four dimensions of interest: (i) point of integration, i.e. in which point the information coming from text and structured data sources was integrated during the process of automatic extraction of knowledge graphs; (ii) integration goal, i.e. if the goal is to expand the knowledge graph or validate its current content; (iii) format of the structured data, and (iv) format of the final knowledge graph, i.e. the type of knowledge graph expected as the output of the the extraction process using both text and structured data source.

Three **point of integration** were considered in our analysis. *Pre-construction* (Figure 2a) refers to the enrichment of the textual documents with information from the structured data source before they are provided as input to the knowledge graph construction algorithm. Integration that happens *during construction* (Figure 2b) assumes that both textual and structured data sources are not linked in advance and, instead, their linking will be performed during the process of automatic knowledge graph extraction. Last, *post-construction* (Figure 2c) stands for the connection between the output of the automatic extraction of knowledge graph from text and information coming from the structured data source.

The joint use of text and structured data sources can be used for two different **integration goals**: (i) knowledge graph completion, where data from both sources are combined to extend the knowledge graph, and (ii) knowledge graph validation, in which one data source is used to validate the information from the other data source.

Since enterprise environments work with heterogeneous types of data, the third dimension of interest relates to the **format of the structured data**. When analysing this dimension we are simplifying our categorisation by assuming that any data source that can be represented by an entity-relation diagram can be considered, or converted to, a graph.

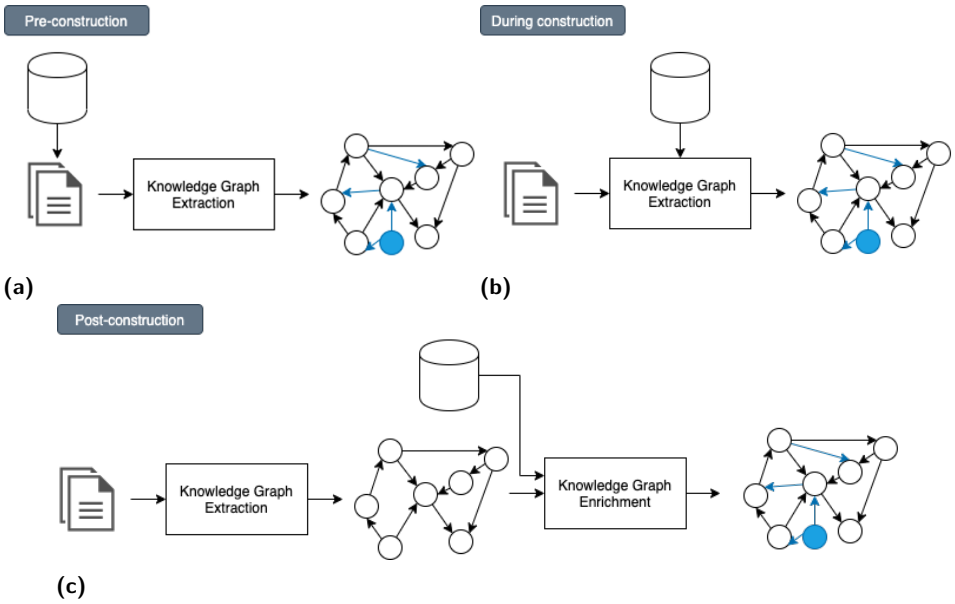


Figure 2 Integration of textual and structured data sources to enrich the automatic construction of knowledge graphs (a) pre-construction, (b) during construction, and (c) post-construction.

Last, given that there are multiple definitions of knowledge graph in the literature, we also analyse the **format of the output knowledge graph**. Four types of knowledge graph were considered: (i) term taxonomy, i.e. a knowledge graph only contains taxonomic relations and where vertices are terms, (ii) topic taxonomy, similar to term taxonomies but where vertices are represented by sets of terms, (iii) labeled graphs, in which relations receive a label representing a relation type (e.g. usedFor, preRequisite), and (iv) ontologies, where vertices and relations have a formal logical representation associated with them.

5 Results and Discussion

Based on the list of seed papers and their references, we start our analysis based on a set of 131 papers. From those, 97 papers were removed due to our exclusion criteria, resulting in one seed paper and further 34 papers analysed. Table 2 presents the categorisation of each paper and in the next sections we present the results of our analysis according to each of the dimensions chosen for literature review.

5.1 Point of Integration

The approaches used for **pre-construction** leverage the knowledge graph extraction algorithm in one of three ways: (i) by combining both textual and structured information into a single semantic space via the use of text embeddings and motifs [32], (ii) by generating a profile for each term using the metadata of the documents associated to the term (e.g. popularity of a document) [44], then using this profile to determine where the term fits in the structure of the final knowledge graph, and (iii) by using the structured data source to generate an initial graph that connects the different terms from text [42].

Regarding the integration **during construction**, the structured data source is provided as training data for the detection of relations between terms extracted from text. This detection is based on either: (i) relation classification, or (ii) relation prediction. In relation

Table 2 Papers categorised according to our dimensions of analysis.

Dimension of Analysis		Paper References
Point of integration	Pre-construction	[32, 42, 44]
	During construction	[39, 19, 12, 13, 41, 22, 40, 37, 16, 34, 28, 3, 14, 15]
	Post-construction	[7, 35, 9, 43, 46, 36, 17, 38, 8, 6, 20, 18, 1, 4, 10, 30, 27]
Integration goals	Completion	[21, 7, 39, 19, 12, 13, 41, 22, 40, 37, 16, 35, 9, 34, 28, 3, 14, 15, 43, 46, 36, 17, 38, 8, 6, 1, 4, 10, 24, 30, 27, 42, 44]
	Validation	[7, 39, 19, 12, 13, 41, 22, 40, 9, 14, 20, 18]
Format of structured data	Table	[32]
	Key-value pairs	[42, 44]
	Graph	[32, 7, 39, 19, 12, 13, 41, 22, 40, 37, 16, 35, 9, 34, 28, 3, 14, 15, 43, 46, 36, 17, 38, 8, 6, 20, 18, 1, 4, 10, 24, 30, 27]
Format of the output knowledge graph	Term taxonomy	–
	Topic taxonomy	[32, 42, 44]
	Labeled graph	[7, 39, 19, 12, 13, 41, 22, 40, 37, 16, 9, 34, 28, 3, 14, 15, 43, 46, 36, 17, 38, 8, 6, 20, 18, 1, 4, 10, 24, 30, 27]
	Ontology	[35]

classification, the types of relations existent in the structured data source are used to classify the relations existent between any two terms from text. In the reviewed literature, this is achieved by using neural networks ([3, 14, 15, 34]), or probabilistic methods ([28]). In contrast, in relation prediction, the goal is to identify what is the target term (or entity) to which any single term extracted should be related to. The literature focused on the use of neural networks using representations that are: (i) based on embeddings [39, 19, 12, 13, 41, 22, 40], or (ii) based on tensors [16, 37].

In **post-construction integration**, the knowledge graph built from text can be integrated with other knowledge graphs by: (i) graph alignment, where knowledge graphs are linked to each other but are still kept as separate entities [10, 17, 1, 35, 30, 7, 8, 46, 36, 6, 43, 8, 38], (ii) graph fusion, where the knowledge graphs have their terms and structures merged into a single knowledge graph [4, 9], or (iii) logical inference, where one knowledge graph is used to extract inference rules for expansion of the data in the other knowledge graph [27].

5.2 Integration Goals

We expect the literature to be heavily biased towards the goal of knowledge graph completion, i.e. extending the knowledge graph with information from both text and structured data sources. This is confirmed by our analysis where 33 papers out of 40 focused exclusively on this goal. The papers that focus only on the validation of knowledge graphs are limited to verifying the correctness of entities and relations but do not perform any additional step of correcting the detected errors [18, 20].

5.3 Format of the Structured Data

The literature explores the use of three different formats of structured data: (i) tables, (ii) key-value pairs , and (iii) graphs.

Tables and value-key pairs are provided as metadata to textual documents in pre-construction approaches either by the use of explicit links between entities in the structured data and documents that refer to those entities [32], or by the inference of these links via analysis of user interactions with both data sources [42, 44]. Graphs, on the other hand, are a dominant format in the analysed literature (Table 2), therefore enterprises wishing to integrate text and structured data for knowledge graph construction would have a higher availability of approaches if the structured data source is a graph-like structure.

5.4 Format of the Output Knowledge Graph

The literature analysed have a strong focus on labelled graphs, while the extraction of taxonomies and ontologies is underrepresented. This demonstrates that despite the amount of work on automatic extraction of taxonomies from text or ontology generation from structured data sources, it does not seem to be a common practice to integrate the two types of data sources when generating taxonomies or ontologies.

6 Conclusion

The goal of this paper is to provide knowledge on what is available in the literature for use by enterprises wishing to generate knowledge graphs based on their own internal data sources. For that, we present a preliminary literature review that investigates approaches used for the integration of textual and structured data sources in the process of automatic knowledge graph construction. Our analysis was based on: (i) point of integration (before, during or after knowledge graph construction), (ii) the goal for integrating sources, (iii) the format of the structured data source, and (iv) the structure of the constructed knowledge graph. Based on this analysis we conclude that, enterprises have a range of approaches available if aiming at the generation of a labelled knowledge graph that aggregates data from both textual and structured sources, where the structured data source used has a graph-like structure and the integration between textual and structured source is done only after a knowledge graph has been extracted from text (what we name post-construction integration). Meanwhile, the integration of data sources before they are used for automatic knowledge graph construction, as well as the use of tables or key-value pairs as structured data sources are still areas with possibility for further research.

7 Lessons Learned and Future Work

Many lessons can be drawn from this specific analysis in terms of our conceptual framework, survey analysis and findings.

Our categorization, while specific, has shown to be useful in classifying available approaches for constructing a domain-specific knowledge graph by; (i) categorizing similar approaches based on the selected dimensions, and (ii) displaying the patterns that influence the decision to adapt a specific variation of each dimension as discussed in the results section. The value of this classification is that it provides enterprises with a clear set of approaches for constructing a domain-specific knowledge graphs from structured and unstructured data sources. As a result, this survey is a step forward in understanding the possible solutions for generating domain-specific enterprise knowledge graph. It also assists enterprise practitioners who may prefer one approach over the another due to constrains in resource, time, and cost.

From the survey perspective, there is an opportunity to future investigate the research and the application of knowledge graph in enterprise domain. Future work will include expanding the survey to a systematic review with keyword-based seed papers. We envision this as a large-scale study that will examine the enterprise knowledge graph integration from different perspectives and demonstrate use cases from various application domains.

From the perspective of survey results, there are a range of options for generating knowledge graphs by aggregating structured and unstructured sources. According to our findings, integration using graph-like structure is a popular approach in comparison to tables and key-value pairs. The later formats are currently in the tentative stage as outlined in the results section. There is also an equal interest in integrating resources during or after the construction of the knowledge graph, as opposed to leveraging resources before the knowledge graph is created. In terms of integration goals, most research focuses on completing a knowledge graph, whereas only a few focus on using data sources to validate a populated knowledge graph. Finally, the majority of work is designed to generate a labelled graph as an output, with a few recent work focusing on topic knowledge graph for classification purposes.

References

1 Alessio Palmero Aprosio, Claudio Giuliano, and Alberto Lavelli. Automatic expansion of dbpedia exploiting wikipedia cross-language information. In *Extended Semantic Web Conference*, pages 397–411. Springer, 2013.

2 Georgeta Bordea, Els Lefever, and Paul Buitelaar. SemEval-2016 task 13: Taxonomy extraction evaluation (TExEval-2). In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1081–1091. ACL, 2016.

3 Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *NeurIPS*, pages 1–9, 2013.

4 Volha Bryl and Christian Bizer. Learning conflict resolution strategies for cross-language wikipedia data fusion. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 1129–1134, 2014.

5 Hongyun Cai, Vincent W Zheng, and Kevin Chen-Chuan Chang. A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE Transactions on Knowledge and Data Engineering*, 30(9):1616–1637, 2018.

6 Muhao Chen, Yingtao Tian, Kai-Wei Chang, Steven Skiena, and Carlo Zaniolo. Co-training embeddings of knowledge graphs and entity descriptions for cross-lingual entity alignment. In *IJCAI*, pages 3998–4004, 2018.

7 Muhao Chen, Yingtao Tian, Mohan Yang, and Carlo Zaniolo. Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. In *IJCAI*, pages 1511–1517, 2017.

8 Gustavo de Assis Costa and José Maria Parente de Oliveira. Linguistic frames as support for entity alignment in knowledge graphs. In *Proceedings of the 20th International Conference on Information Integration and Web-based Applications & Services*, pages 226–229, 2018.

9 Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–610, 2014.

10 Arnab Dutta, Christian Meilicke, and Simone Paolo Ponzetto. A probabilistic approach for integrating heterogeneous knowledge sources. In *European Semantic Web Conference*, pages 286–301. Springer, 2014.

11 Jérôme Euzenat, Christian Meilicke, Heiner Stuckenschmidt, Pavel Shvaiko, and Cássia Trojahn. Ontology alignment evaluation initiative: Six years of experience. In Stefano Spaccapietra, editor, *Journal on Data Semantics XV*, pages 158–192. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

12 Jun Feng, Minlie Huang, Yang Yang, and Xiaoyan Zhu. Gake: Graph aware knowledge embedding. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 641–651, 2016.

13 Shu Guo, Quan Wang, Bin Wang, Lihong Wang, and Li Guo. Semantically smooth knowledge graph embedding. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 84–94, 2015.

14 Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. Knowledge graph embedding via dynamic mapping matrix. In *Proceedings of the 53rd annual meeting of the ACL and the 7th international joint conference on natural language processing (volume 1: Long papers)*, pages 687–696, 2015.

15 Guoliang Ji, Kang Liu, Shizhu He, and Jun Zhao. Knowledge graph completion with adaptive sparse transfer matrix. In Dale Schuurmans and Michael P. Wellman, editors, *AAAI*, pages 985–991. AAAI Press, 2016.

16 Seyed Mehran Kazemi and David Poole. Simple embedding for link prediction in knowledge graphs. In *NeurIPS*, pages 4289–4300, 2018.

17 Chao Kong, Ming Gao, Chen Xu, Yunbin Fu, Weining Qian, and Aoying Zhou. EnAli: entity alignment across multiple heterogeneous data sources. *Frontiers of Computer Science*, 13(1):157–169, 2019.

18 Jens Lehmann, Daniel Gerber, Mohamed Morsey, and Axel-Cyrille Ngonga Ngomo. Defacto-deep fact validation. In *International semantic web conference*, pages 312–327. Springer, 2012.

19 Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In Blai Bonet and Sven Koenig, editors, *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 2181–2187. AAAI Press, 2015.

20 Shuangyan Liu, M. d'Aquin, and E. Motta. Towards linked data fact validation through measuring consensus. In *LDQ@ESWC*, 2015.

21 Shuangyan Liu, Mathieu d'Aquin, and Enrico Motta. Measuring accuracy of triples in knowledge graphs. In *LDK*, pages 343–357. Springer, 2017.

22 Yuanfei Luo, Quan Wang, Bin Wang, and Li Guo. Context-dependent knowledge graph embedding. In *Proceedings of the 2015 Conference on EMNLP*, pages 1656–1661, 2015.

23 Paul McNamee and Hoa Trang Dang. Overview of the tac 2009 knowledge base population track. In *Text Analysis Conference (TAC)*, volume 17, pages 111–113, 2009.

24 Emir Muñoz, Aidan Hogan, and Alessandra Mileo. Triplifying wikipedia's tables. *LD4IE@ ISWC*, 2013.

25 Hoang Long Nguyen, Dang Thinh Vu, and Jason J Jung. Knowledge graph fusion for smart systems: A survey. *Information Fusion*, 61:56–70, 2020.

26 Heiko Paulheim. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web*, 8(3):489–508, 2017.

27 Heiko Paulheim and Simone Paolo Ponzetto. Extending dbpedia with wikipedia list pages. *NLP-DBPEDIA@ ISWC*, 13, 2013.

28 Boya Peng, Yejin Huh, Xiao Ling, and Michele Banko. Improving knowledge base construction from robust infobox extraction. In *Proceedings of NAACL HLT Conference, Volume 2 (Industry Papers)*, pages 138–148, 2019.

29 Bianca Pereira, C. Robin, Tobias Daudert, John P. McCrae, Pranab Mohanty, and P. Buitelaar. Taxonomy extraction for customer service knowledge base construction. In *SEMANTiCS*, 2019.

30 Dominique Ritze, Oliver Lehmberg, and Christian Bizer. Matching html tables to dbpedia. In *Proceedings of the 5th International Conference on Web Intelligence, Mining and Semantics*, pages 1–6, 2015.

31 G. Rizzo, B. Pereira, A. Varga, M. Van Erp, and A.E.C. Basave. Lessons learnt from the Named Entity rEcognition and Linking (NEEL) challengeÂ series. *Semantic Web*, 8(5):667–700, 2017.

32 Jingbo Shang, Xinyang Zhang, Liyuan Liu, Sha Li, and Jiawei Han. Nettaxo: Automated topic taxonomy construction from text-rich network. In *Proceedings of The Web Conference 2020*, pages 1908–1919, 2020.

33 Wei Shen, Jianyong Wang, and Jiawei Han. Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering*, 27(2):443–460, 2014.

34 Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. Reasoning with neural tensor networks for knowledge base completion. In *NeurIPS*, pages 926–934. Citeseer, 2013.

35 Fabian M Suchanek, Serge Abiteboul, and Pierre Senellart. Paris: probabilistic alignment of relations, instances, and schema. *Proceedings of the VLDB Endowment*, 5(3):157–168, 2011.

36 Zequn Sun, Wei Hu, and Chengkai Li. Cross-lingual entity alignment via joint attribute-preserving embedding. In *International Semantic Web Conference*, pages 628–644. Springer, 2017.

37 Yi Tay, Anh Tuan Luu, Siu Cheung Hui, and Falk Brauer. Random semantic tensor ensemble for scalable knowledge graph link prediction. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 751–760, 2017.

38 Lucy Wang, Chandra Bhagavatula, Mark Neumann, Kyle Lo, Chris Wilhelm, and Waleed Ammar. Ontology alignment in the biomedical domain using entity definitions and context. In *Proceedings of the BioNLP 2018 workshop*, pages 47–55, 2018.

39 Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge graph embedding by translating on hyperplanes. In Carla E. Brodley and Peter Stone, editors, *AAAI*, pages 1112–1119, 2014.

40 Zhigang Wang, Juanzi Li, Zhiyuan Liu, and Jie Tang. Text-enhanced representation learning for knowledge graph. In *IJCAI*, pages 4–17, 2016.

41 Ruobing Xie, Zhiyuan Liu, and Maosong Sun. Representation learning of knowledge graphs with hierarchical types. In *IJCAI*, pages 2965–2971, 2016.

42 Xiaoxin Yin and Sarthak Shah. Building taxonomy of web search intents for name entity queries. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, page 1001–1010. ACM, 2010.

43 Youmin Zhang, Li Liu, Shun Fu, and Fujin Zhong. Entity alignment across knowledge graphs based on representative relations selection. In *ICSAI*, pages 1056–1061. IEEE, 2018.

44 Yuchen Zhang, Amr Ahmed, Vanja Josifovski, and Alexander Smola. Taxonomy discovery for personalized recommendation. In *Proceedings of the 7th ACM international conference on Web search and data mining*, pages 243–252, 2014.

45 Xiaojuan Zhao, Yan Jia, Aiping Li, Rong Jiang, and Yichen Song. Multi-source knowledge fusion: a survey. *World Wide Web*, 23(4):2567–2592, 2020.

46 Hao Zhu, Ruobing Xie, Zhiyuan Liu, and Maosong Sun. Iterative entity alignment via joint knowledge embeddings. In *IJCAI*, volume 17, pages 4258–4264, 2017.

An Ontology for CoNLL-RDF: Formal Data Structures for TSV Formats in Language Technology

Christian Chiarcos
Applied Computational Linguistics Lab, Goethe University Frankfurt, Germany

Maxim Ionov
Applied Computational Linguistics Lab, Goethe University Frankfurt, Germany

Luis Glaser
Applied Computational Linguistics Lab, Goethe University Frankfurt, Germany

Christian Fäth
Applied Computational Linguistics Lab, Goethe University Frankfurt, Germany

Abstract

In language technology and language sciences, tab-separated values (TSV) represent a frequently used formalism to represent linguistically annotated natural language, often addressed as "CoNLL formats". A large number of such formats do exist, but although they share a number of common features, they are not interoperable, as different pieces of information are encoded differently in these dialects.

CoNLL-RDF refers to a programming library and the associated data model that has been introduced to facilitate processing and transforming such TSV formats in a serialization-independent way. CoNLL-RDF represents CoNLL data, by means of RDF graphs and SPARQL update operations, but so far, without machine-readable semantics, with annotation properties created dynamically on the basis of a user-defined mapping from columns to labels. Current applications of CoNLL-RDF include linking between corpora and dictionaries [28] and knowledge graphs [36], syntactic parsing of historical languages [12, 11], the consolidation of syntactic and semantic annotations [8], a bridge between RDF corpora and a traditional corpus query language [24], and language contact studies [6].

We describe a novel extension of CoNLL-RDF, introducing a formal data model, formalized as an ontology. The ontology is a basis for linking RDF corpora with other Semantic Web resources, but more importantly, its application for transformation between different TSV formats is a major step for providing interoperability between CoNLL formats.

2012 ACM Subject Classification Information systems → Graph-based database models; Computing methodologies → Language resources; Computing methodologies → Knowledge representation and reasoning

Keywords and phrases language technology, data models, CoNLL-RDF, ontology

Digital Object Identifier 10.4230/OASIcs.LDK.2021.20

Supplementary Material *Dataset (Ontology)*: `https://doi.org/10.5281/zenodo.4361476`

Funding This work was funded by the project "Prêt-à-LLOD" within the European Union's Horizon 2020 research and innovation programme under grant agreement no. 825182, as well as the project "Linked Open Dictionaries" (LiODi), funded within the eHumanities program of the German Ministry of Education and Science (BMBF, 2015-2021).

1 Motivation: Incompatible TSV formats

The automated analysis of natural language requires different, and often, complex steps of processing, traditionally organized in a pipeline architecture. Depending on the specific goals, this does include designated modules for standard tasks such as sentence splitting, tokenization, part-of-speech labelling, lemmatization, morphological analysis, named entity

recognition, word sense disambiguation, entity linking, chunking, syntactic parsing, semantic parsing, coreference analysis, but also extend to more research-intense challenges such as discourse parsing, zero anaphora resolution or implicit semantic role labelling. For each of these processing steps, numerous implementations and data sets to train your own classifiers upon are available, normally, the formats they use only support information that is relevant to their specific annotation task. They do, however, usually follow common conventions, as both data sets and reference implementations often originate from long-standing series of shared tasks, and for the family of formats under consideration here, these are also shared with other applications in corpus linguistics and digital lexicography.

Tab-separated values (TSV) are a frequently used formalism to represent linguistically annotated natural language, e.g., in the long-standing series of Shared Tasks of the Conference of Natural Language Learning (CoNLL), recent initiatives on the creation of corpora and tools with cross-linguistically applicable ("universal") annotations [31, Universal Dependencies, UD], [27, UniMorph], [2, Universal Propositions], or in computational lexicography and corpus linguistics [13, Corpus Workbench], [26, Sketch Engine]. Many such "CoNLL" formats exist, but although they share a number of common features (e.g., one word per line, empty line to mark sentence breaks, comments after #), they are not interoperable with each other, as different pieces of information are represented differently in different dialects, e.g., placed in different columns or spread over multiple columns in one format, but consolidated into one in another.

CoNLL-RDF [7] is a set of tools introduced to facilitate processing and transforming CoNLL and other TSV formats in a serialization-independent way: On the basis of a user-provided mapping from columns to labels (properties), sentence by sentence (blocks of annotations separated by empty lines), tab-separated data is transformed to RDF graphs in accordance with the CoNLL-RDF data model.[1] Annotations can then be manipulated using SPARQL Update operations and serialized in TSV, RDF or XML formats. Unlike CSV2RDF [20], R2RML [15], and related general-purpose technology for mapping tabular data to RDF, CoNLL-RDF provides *linguistic* data structures: The CoNLL-RDF data model uses the NLP Interchange Format [21, NIF] to encode sentences, words and sequential relations between these, and extends it with properties for the annotation of words, syntactic dependencies and semantic roles. First introduced in 2017, this technology is now being used in a number of projects in NLP [1], knowledge engineering [19], linguistics [29] and Digital Humanities [22].

We describe a novel extension of CoNLL-RDF technology, introducing a formal data model. So far, CoNLL-RDF used a shallow approach to semantics, where annotation properties were created dynamically on the basis of a user-defined mapping from columns to labels (property names), without any machine-readable semantics: CoNLL-RDF representations were data-driven and unrestricted, so that the same information could be found under different properties, etc. It is, however, essential to provide machine-readable semantics for these properties as individual CoNLL dialects record this information differently.

The CoNLL-RDF ontology provides machine-readable semantics for an inventory of CoNLL properties (and classes) for a growing collection of about two dozen CoNLL and related formats currently used in language technology. In addition, a mapping between CoNLL properties and columns provides a formal, and machine-readable definition of the

[1] Even though sentence by sentence transformation creates a computational overhead, it prevents from having memory issues while processing large amounts of text. A detailed discussion of this design decision can be found in the original paper.

respective formats. Using this information, we provide a mapping between different TSV formats. A user is only required to specify input and output formats (say, CoNLL-U and CoNLL-X). Using the column mappings defined in the CoNLL-RDF ontology, we derive a transformation workflow that retrieves information from source columns, (optionally) transforms it and allocates them to the corresponding columns in the target format. This workflow is then executed using the Flexible Integrated Annotation Engineering (Fintan) platform [18].

The CoNLL-RDF ontology, introduced with this paper, adds machine-readable semantics for existing datasets encoded as CoNLL-RDF and provides the basis for linking RDF corpora with other Semantic Web resources. With the ontology, the relations between 24 TSV formats have be made explicit in a machine-readable way, and it now becomes possible to (a) automatically transform one TSV format into another, resp. (b) to assert/infer that a particular format cannot be automatically transformed into another. Aside from the ontology, we introduce CoNLL-Transform, a converter that uses the CoNLL-RDF ontology to bootstrap automated conversion routines.

In the context of transformation, the CoNLL-RDF ontology serves two main purposes: It provides a mapping from columns to properties, and it defines standard identifiers (URIs) for these properties. Even though this aspect is beyond the scope of the current paper, this is the basis for develop transformers that are capable to perform more complex operations, e.g., to derive CoNLL-2004 chunking information from a CoNLL-U dependency parse.

2 Background: CoNLL-RDF

Natural language processing (NLP) and knowledge graphs are two critical areas in the developmentof language technologies. Building bridges between the two bears potential to enable progress in both. CoNLL-RDF has been designed to serve as such a technological bridge, enabling researchers to easily go back-and-forth between popular one-word-per-line TSV formats used in language technology, and SPARQL and Semantic Web technologies used in knowledge engineering. CoNLL-RDF refers to a library that allows parsing each sentence from a CoNLL-TSV data stream (together with its context) into a separate RDF graph, to manipulate and to enrich it with SPARQL and reasoning technologies, and to serialize the result back to Turtle,[2] to (any) TSV format or to a number of other common formats used in language technology.

The CoNLL-RDF library is part of the Flexible Integrated Annotation Engineering (Fintan) platform [18], but also distributed individually. It is available as open source (Apache license 2.0) from our Github repository.[3]

2.1 One-word-per-line TSV formats in language technology ("CoNLL")

One-word-per-line formats, especially tab-separated value (TSV) formats have been a popular choice in a variety of applications for more than three decades. The fields of use include digital lexicography (SketchEngine [26]), corpus linguistics (Corpus Workbench/CWB [16]), natural language processing (TreeTagger [35]), and as an exchange format in a variety of different corpora projects. These formats enjoy continued and rising popularity because TSV allows for flexible encoding of any kind of word-level annotations, they provide an ideal

[2] That is, a canonical TTL representation that emulates the structure of CoNLL/TSV.
[3] `https://github.com/acoli-repo/conll-rdf`

middle ground between being machine-processable and human-readable, and they can be easily extended by creating additional columns with new annotations. As a result, TSV formats have become a de-facto standard in exchanging NLP data.

The listing below is an example from the 2005 Shared Task of the SIGNLL Conference on Computational Natural Language Learning (CoNLL-05):

```
#  WORDS          NE   POS  PARTIAL_SYNT       PARSE
   The             *   DT    (NP*   (S*        (S (NP *
   spacecraft      *   NN       *)    *               *)
   faces           *   VBZ   (VP*)    *           (VP *
   a               *   DT    (NP*     *           (NP *
   ...
```

Here, the first column contains the word, the second column contains named entity annotation, the third contains part-of-speech information. The following columns contain different forms of syntax annotation: The `PARTIAL_SYNT` column has two subcolumns, where the first subcolumn contains nominal and verbal chunks, and the second subcolumn contains sentence chunks. The `PARSE` column contains a full parse in accordance with the Penn Treebank [30].

Subsequently, the use of TSV formats to exchange linguistic data has since extended its spread beyond the CoNLL Shared Task and inspired novel corpora formats, e.g. the CoNLL-U format by Universal Dependencies has been created independent of the conference; however adheres and extends standards already motivated by the CoNLL Shared Task. In this paper we follow this convention by referring to all one-word-per-line TSV formats as CoNLL-TSV.

2.2 Words and sentences

CoNLL-RDF can transform any CoNLL-TSV dialect into a CoNLL-RDF representation, apply SPARQL updates to the transformed sentences and re-serialize the representations into RDF or TSV as defined by the user.

The CoNLL-RDF vocabulary builds on a minimal fragment of the NIF data model: Each word (row) is represented as a `nif:Word`, and connected to the following word of the same sentence by the `nif:nextWord` property. Each sentence (sequence of rows not interrupted by an empty line) is represented as a `nif:Sentence`, and connected to the following sentence by the `nif:nextSentence` property. Words can be organized in a dependency tree (using the `conll:HEAD` property), and for words that do not have a parent in the dependency annotation (incl. formats where no `HEAD` column is given), are linked by `conll:HEAD` to the respective `nif:Sentence`.

For representing annotations, the CoNLL-RDF toolchain uses column labels provided by the user in order to associate each column with a novel property in the `conll` namespace. CoNLL-TSV can be transformed into CoNLL-RDF using ad hoc labels, but these are not backed by a formal ontology. In general, these column labels were simply treated as such; with the sole exception of semantic role annotations in the form of `PRED_ARGs` and the `HEAD` column. Both of these carry special semantics and are handled specially during conversion.

In real-world applications, e.g., the creation of novel forms of syntactic-semantic annotation [9] or experimental forms of syntactic parsing [12], where specialized data structures are required, a more constrained view is taken, and consistent labels should be used throughout the project – but so far, CoNLL-RDF provides no way to facilitate interoperability and interpretability of column labels across different annotation projects.

2.3 Tree extension

One-word-per-line formats originally provide no base vocabulary for representing annotations that span beyond more than one token, and different extensions have been developed throughout the CoNLL shared tasks. These include the IOB(ES) annotation for non-recursive spans introduced with the CoNLL-00 Shared Task [34], the bracket notation of the Penn Treebank [30], and the application of XML, resp., SGML markup *between* the word-level annotations (as used by TreeTagger, the Sketch Engine and the Corpus Workbench).

We illustrate tree structures with the bracket notation from the sixth column (`PARSE`) of the CoNLL-05 example given above: The original CoNLL-RDF implementation represented these structures as plain string literals, without analyzing their internal structure, i.e., as `conll:PARSE "(S ( NP *"`, etc. Processing such data with SPARQL is possible but cumbersome, as the strings need to be decoded before their content can be analysed. In essence, a user would need to write a CFG parser in SPARQL – and this is possible, but slow. Chiarcos and Glaser [10] thus extended the CoNLL-RDF library with routines for the native parsing and serialization of such structures. In order to avoid the introduction of ad hoc data structures into the CoNLL-RDF data model, the internal representation of phrases is grounded in the POWLA vocabulary [4].

POWLA provides an OWL2/DL formalization of the Linguistic Annotation Framework (LAF) as described by [23, 25]. LAF provides generic data structures for representing *any* kind of linguistic annotation. In particular, this includes a separation of positions and spans in the primary data (represented as anchors and regions, comparable to the target element of Web Annotation, or to instances of `nif:String`) and annotations (represented as nodes and edges/relations, comparable to annotations, resp., their bodies in Web Annotation; NIF, instead, recommends to model annotations as `nif:String` objects).[4] POWLA does not provide a formalization of anchors and regions, but builds on other vocabularies for this purpose, most notably NIF and Web Annotation [14]. Accordingly, the CoNLL-RDF tree extension uses POWLA data structures primarily to represent data structures above the level of the token (word), and for these, POWLA allows to define an annotation graph independent of the sentence and word structure imposed by CoNLL.

In CoNLL-RDF, only a minimal fragment of the POWLA vocabulary is used, partially with slightly more constrained definitions than in POWLA originally:

powla:Node Within CoNLL-RDF, every `nif:Word` is a `powla:Node`. Other nodes in CoNLL-RDF are recursively defined as a `nif:Word` or any grouping of `powla:Node`s.

powla:hasParent property pointing from an element to the phrase (or other aggregate node) that contains it.

powla:next To facilitate navigation in POWLA graphs, adjacent `powla:Node`s that share the same parent (and only these) should be connected by the `powla:next` property. A `powla:Node` may have multiple `powla:next` properties relative to different parent nodes, e.g., if multiple levels of syntax annotation are provided as in the three levels of syntax annotation of CoNLL-05.

powla:Relation for representing labelled (annotated) edges, POWLA allows to reify relations between a source (`powla:hasSource`) and a target node (`powla:hasTarget`). POWLA relations are not automatically generated during the process of parsing TSV formats, but can be created and used by subsequent SPARQL Update operations.

[4] Note that NIF 2.0 also introduced a `nif:Annotation` object, but `nif:Phrase`, etc., are still defined as `nif:String` subclasses.

When parsing the bracketing notation, the original column name is maintained as a datatype property, but applied to the POWLA node rather than the `nif:Word`. The value of this property is the label of the corresponding phrase. For the CHUNK column (column 4) in the example, three POWLA nodes are created:

- a node containing *The spacecraft*, with `conll:CHUNK 'NP'`
- a node containing *faces*, with `conll:CHUNK 'VP'`
- a node containing *a ...*, with `conll:CHUNK 'NP'`

SENTENCE (column 5) and PARSE (column 6) are processed analoguously.

3 CoNLL-RDF ontology

We designed the CoNLL-RDF ontology to capture the semantic and structural dependencies of annotations in TSV formats and tree extensions, with a focus on properties, as the basic structures of annotation are defined in external vocabularies, i.e., NIF (words and sentences) and POWLA (other units of annotation).

The ontology consists of two components: (1) classes, properties and axioms used to define formats (Fig. 1), and (2) the machine-readable description of existing CoNLL and related formats. The namespace prefix is `conll:`.

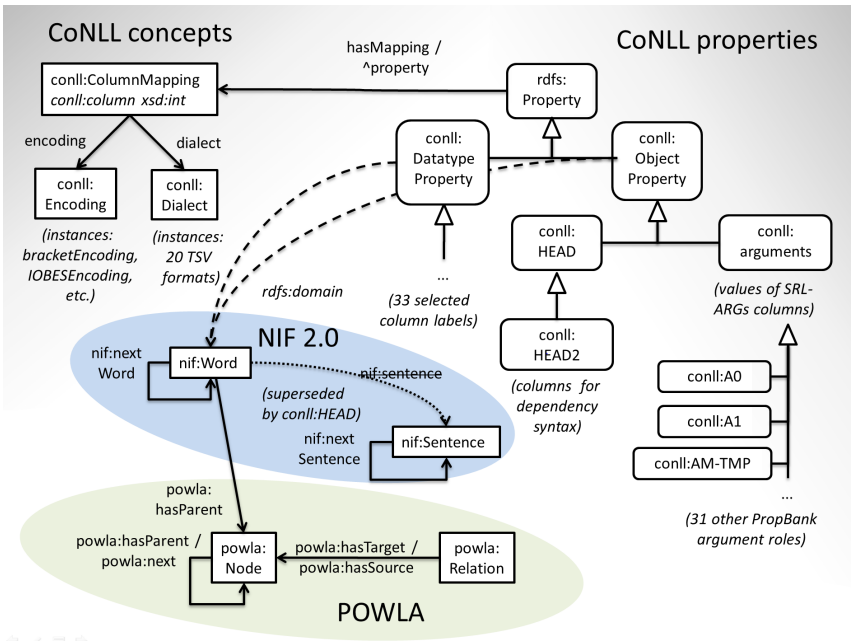


Figure 1 CoNLL-RDF Ontology: Classes and properties of CoNLL and external vocabularies.

As for CoNLL properties, the ontology provides a catalog of 33 datatype properties, with human-readable descriptions and labels as used in previous literature, and organized in an inheritance structure. The column label WORD, used in CoNLL Shared Tasks until 2005, does, for example, roughly correspond to the column label FORM, but the latter is a generalization, so that `conll:WORD` is a `:subPropertyOf conll:FORM`.

As for object properties, their creation is triggered by conventions in the CoNLL-RDF tool chain: HEAD contains the ID (or sentence position) of the syntactic head, and `conll:HEAD` refers to the head *or* to the sentence (if no head does exist). The column label *PRED_ARGS*

is used for semantic role annotations, where every predicate in a sentence triggers the creation of a subsequent argument column that annotates all words for their respective semantic role relative to this particular predicate. Here, the CoNLL properties are not generated from user-provided labels, but from the labels used in the annotation, e.g., `conll:A0` for the agent argument of a transitive verb. The CoNLL ontology thus contains the full inventory of PropBank roles.

We provide `conll:DatatypeProperty` and `conll:ObjectProperty` as subproperties of `rdf:Property` – not because we want to replicate OWL semantics, but because we restrict their domain to `nif:Word`s and `powla:Node`s.

As for classes, the CoNLL ontology does not introduce data categories, but concepts for metadata only (the mapping from CoNLL properties to different formats). It does, however, refine NIF and POWLA concepts with a more constrained definition than in their original vocabularies. As an example, a `nif:Word` within CoNLL-RDF must be an instance of `powla:Node`.

Each CoNLL-TSV and related TSV format is represented by an individual of the `conll:Dialect` type. A minimal dialect definition consists of a name (`rdfs:label`) and a link to documentation (`rdfs:isDefinedBy`).

A dialect *may* be used in one or more `conll:ColumnMapping`s. A column mapping links a CoNLL property (`conll:property`) with a particular column position (`conll:column`) in a particular format (`conll:dialect`). Any CoNLL property can be related to multiple mappings. Each relation then describes a mapping for a specific property in a specific dialect. This allows to represent data independent of the exact dialect. Instantiations of both property types will be represented by a column in the TSV file or by a `conll:column` property in CoNLL-RDF. In different TSV dialects, these will sit in different columns, considering their index. In addition, the column mapping can define the encoding strategy of POWLA nodes by means of the `conll:encoding` property. Note that the same property can be encoded in different ways, as shown for the bracket notation above and the IOBES encoding below.

With the classes and properties introduced above, we are able now to model CoNLL data in CoNLL-RDF, independent of the dialect in which it was originally encoded. As part of the ontology, we provide formal data structures and properties for 22 CoNLL-TSV and related TSV dialects. This includes all CoNLL Shared Task TSV formats until 2018, CoNLL-U, UniMorph, several PropBank formats, the formats of the Open Multilingual WordNet initiative, SketchEngine, Corpus WorkBench, TreeTagger, and the format of a series of Shared Tasks on Translation Inference Across Dictionaries (TIAD). We illustrate this below for the CoNLL-00 format:

```
The             DT    B-NP
spacecraft      NN    I-NP
faces           VBZ   B-VP
a               DT    B-NP
...
```

The CoNLL-00 columns are WORD, POS and CHUNK, for a Shared Task on chunking (shallow syntax). Note that the CoNLL-RDF tree extension renders the content of the CHUNK column *exactly* in the same way as the CHUNK information from the CoNLL-05 format described in Section 2.1. In the CoNLL-RDF ontology, we provide the full description of the CoNLL-00 format:

```
:CoNLL-00 a :Dialect;
rdfs:label "CoNLL-00 format";
rdfs:isDefinedBy <https://www.clips.uantwerpen.be/conll2000/chunking/>.

# first column (for CoNLL-00): WORD
:WORD rdfs:subPropertyOf :FORM; rdfs:label "WORD";
  rdfs:comment "Word form in an annotated text ..."@en;
  :hasMapping [ a :ColumnMapping; :column "1"^^xsd:int ; :dialect :CoNLL-00 ] ;
  :hasMapping [ a :ColumnMapping; :column "4"^^xsd:int ; :dialect :CoNLL-11 ] . # etc
```

```
# second column: POS
:POS rdfs:subPropertyOf :DatatypeProperty; rdfs:label "POS", "POSTAG", "TAG";
rdfs:comment "Fine-grained part-of-speech tag ..."@en;
:hasMapping [ a :ColumnMapping; :column "2"^^xsd:int; :dialect :CoNLL-00 ] ;
:hasMapping [ a :ColumnMapping; :column "3"^^xsd:int; :dialect :CoNLL-05 ] . # etc

# third column: CHUNK
:CHUNK rdfs:subPropertyOf :DatatypeProperty ; rdfs:label "CHUNK";
rdfs:comment "The chunk tags contain the name of the chunk type,
              for example I-NP ..."@en;
:hasMapping [ a :ColumnMapping; :column "3"^^xsd:int; :dialect :CoNLL-00 ] ;
:hasMapping [ a :ColumnMapping; :column "4"^^xsd:int; :dialect :CoNLL-05 ] . # etc
```

The listing only provides a partial view of the column mappings beyond CoNLL-00, illustrated for one example per CoNLL property. Also, these are slightly simplified, as they do not specify the actual encoding in CoNLL.

Although different `conll:Dialects` may share a `conll:COLUMN` or even the `conll:ColumnMapping`, the textual representation in CoNLL might differ, e.g. phrase structure might be encoded in IOBES or in a bracket notation. This information is encoded by an instance of `conll:Encoding`; `conll:iobesEncoding` resp. `conll:bracketEncoding` in this case.

As an example, CoNLL-05 does contain the same (plus other) annotations as CoNLL-00, but the chunk information (first `PARTIAL_SYNT` column, column 4) uses the bracketing notation of the Penn Treebank (equivalent with the CHUNK information from our CoNLL-00 sample).

The (abbreviated) entry of the property `conll:CHUNK` is presented below.

```
:CHUNK rdfs:subPropertyOf :DatatypeProperty ; # ...
    :hasMapping [
     :encoding :iobEncoding;
     a :ColumnMapping;
     :column "3"^^xsd:int;
        :dialect :CoNLL-00, :CoNLL-01, :CoNLL-03, :CoNLL-04];
    :hasMapping [
     :encoding :bracketEncoding;
     a :ColumnMapping;
     :column "4"^^xsd:int; :dialect :CoNLL-05 ].
```

The `conll:DatatypeProperty conll:CHUNK` sits in the third column in the CoNLL-00, -01, -03 and -04 dialects. In CoNLL-05, the `conll:CHUNK` property moves to the fourth column, the position is encoded using the `conll:column` property. The formats also differ in their encoding: The `conll:CHUNK` column in CoNLL-00, -01, -03 and -04 was encoded using an IOB-schema.[5] In CoNLL-05 however, the `conll:CHUNK` column was not only moved but encoded using a PTB-style annotation, marked with the `conll:encoding` property.

4 CoNLL-Transform: Ontology-based transformation

The Flexible Integrated Annotation Engineering (Fintan) platform is a recently introduced additional abstraction layer on the existing CoNLL-RDF library [18]. While CoNLL-RDF focuses on the transformation of CoNLL corpora, Fintan broadens the scope towards integrating support for other data formats, such as OntoLex-Lemon for lexica by allowing to easily integrate and run existing converters in complex pipelines. It furthermore adds a graphical workflow manager to build, assess and run these pipelines.

[5] Simplified IOBES encoding, using B- (begin of a single-token or multi-token sequence), I- (middle or end of a multi-token sequence), O (no annotation).

The Fintan API distinguishes different types of transformation modules for which CoNLL-RDF provides designated implementations:

- *Loader* modules may consume any type of input data and must write back RDF data. The CoNLL-RDF `CoNLLStreamExtractor` serves as a Loader module which transforms CoNLL into CoNLL-RDF.
- *Update* modules transform a stream of segmented RDF data on multiple threads. Each Update module is defined by the resources and graphs it requires and an iteration over SPARQL update scripts. CoNLL-RDF provides a `CoNLLRDFUpdater` class that implements the Update function.
- *Writer* modules create serializations of the transformed data. In CoNLL-RDF, the `CoNLLRDFFormatter` functions as a writer module that yields RDF serializations (Turtle, canonical CoNLL-RDF), TSV output and other representations. In the context of Fintan, other existing transformation tools may be mapped as Loaders or Writers.

4.1 Transforming CoNLL dialects

We provide CoNLL-Transform as a command-line tool to generate Fintan transformation workflows directly from the CoNLL-RDF ontology. The code and its integration with the Fintan infrastructure will be published under the Apache 2.0 license along with the publication of this paper.

CoNLL-Transform takes three parameters, source format (e.g., CoNLL-00), target format (e.g., CoNLL-05) and the CoNLL-RDF ontology (or a replacement that provides alternative column mappings, etc.). If the format identifiers match the (local names of) `conll:Dialects` in the ontology, we retrieve the corresponding column mappings (and the encoding specification) to derive the corresponding Fintan Loader and Writer configurations, either as a JSON configuration file, or in the form of a shell script. The combination of a particular Loader and a particular Writer already allows the reordering of columns, but moreover, also to switch from one way of phrase-level encoding (say, IOBES) to another (say, bracket notation) – if specified in the ontology.

4.2 Mapping strategies

Most CoNLL-TSV formats will not provide fully equivalent content. CoNLL-Transform also produces a protocol that lists target format properties (columns) not found in the source (which will be replaced by the empty annotation _), as well as source format properties (columns) not expressible in the target format (which will be omitted). In addition to repositioning columns, changing their encoding, and identifying mismatches between formats, CoNLL-Transform uses the subsumption hierarchy of the CoNLL-RDF ontology to derive heuristic mappings of column names.

We employ the following ranking of mapping strategies:

maintain **if** a CoNLL property from the target format occurs in the source format, maintain it; **otherwise:**

generalize **if** a CoNLL property from the source format (say, `conll:WORD` in CoNLL-00) is a subproperty of a CoNLL property from the target format (say, `conll:FORM` in CoNLL-U), copy the object of the source property to the target property and produce a warning; **otherwise**

specialize **if** a CoNLL property from the source format (say, conll:POS in CoNLL-00) is a superproperty of a CoNLL property from the target format (say, conll:XPOS in CoNLL-U), and the target format property does not already exist, copy the object of the source property to the target property and produce a warning; **otherwise**

skip if no other mapping applies, set target property to empty (`_`)

These mappings are provided as a series of SPARQL Update operations and executed by Fintan before the Writer component is called. Note that this mapping is heuristic. In particular the **specialize** step does introduce a certain amount of errors. Along with `conll:XPOS` (original POS annotation of a Universal Dependencies file), CoNLL-U features another subproperty of `conll:POS`, namely `conll:UPOS`. The property `conll:UPOS` is, however, more constrained than `conll:POS`, and restricted to POS annotation in accordance with the Universal Dependencies tagset. While mapping `conll:POS` to `conll:XPOS` will be correct in most cases, our mapping heuristic will also produce an analoguous mapping from `conll:POS` to `conll:UPOS` which will be incorrect in many cases.

In the future development, we also plan to provide an inventory of SPARQL update scripts for transformations between selected pairs of near-equivalent CoNLL properties. As an example, CoNLL-04 uses two columns to represent predicates in semantic role annotation: PRED_LEMMA contains the lemma of the predicate (e.g., `say`), PRED_FRAMESET contains the sense number (for that particular lemma, e.g., `1`). In CoNLL-08, this information is provided in the PRED column, but concatenated with a separator symbol (`say.01`). Once a SPARQL Update with such a concatenation is provided, it should be applied with preference over the `generalize` and `specialize` mappings.

4.3 Encoding

Aside from the mapping, a key challenge of transforming between different CoNLL-TSV dialects is the encoding of annotations spanning multiple words. We adopt the recent tree extension of CoNLL-RDF: Every span is represented as a `powla:Node`, and linked with the words (or nodes) it contains by means of `powla:hasParent`. These nodes then receive the corresponding annotation.

With the ontology, we can now define which CoNLL dialect uses which encoding strategy, and decode the correct spans and labels from the input data. Likewise, we can use the definition of the output format to trigger a serialization according to another encoding, and thus translate between the IOBES and bracket notations in the listings given above. Note that, however, this is not a transformation of the RDF graph, but only a decoding, resp. encoding instruction executed by Fintan Loader and Writer modules, respectively.

5 Related Community Standards

In this paper, we introduced the CoNLL-RDF ontology as a machine-readable formalization of the CoNLL-RDF data model, in order to facilitate interoperability and transformability between different TSV formats in language technology as well as between these and the knowledge representation / knowledge engineering stack. It is to be noted, however, that the CoNLL-RDF vocabulary does remain extensible, i.e., users can still provide ad hoc labels to generate `conll:` properties, and this feature is very much required. For *established* formats, however, the ontology provides instructions for decoding TSV annotation and for encoding CoNLL-RDF graphs in a TSV format.

It is important, however, that the CoNLL-RDF ontology may also serve a role in the development of vocabularies for linguistic annotations on the web. The development of RDF-based data models for language technologies coincides with a growing trend in publishing language resources (lexicons, corpora, dictionaries, etc) as linked open data (LOD) on the Web [3]. In application to language resources, *linguistic linked open data* (LLOD) is concerned with LOD resources that are *linguistically relevant*, i.e., part of an application or a use case in language technology or the language sciences, cumulating in the emergence of the so-called LLOD cloud.[6] In comparison to conventional means of publishing language resources, LLOD allows for as higher independence from domain-specific data formats or vendor-specific APIs, as well as easier access and re-use of linguistic data by semantic-aware software agents [14].

Prominent vocabularies in the LLOD context include OntoLex-Lemon[7] as the main community standard for ontology lexicalization and representing lexical data in RDF. For representing linguistic annotations, however, no single consensus vocabulary has emerged so far, but instead, incompatible and competing specifications. Most notably, these include Web Annotation [33, WA], the NLP Interchange Format [21, NIF] and the LAPPS Interchange Format [37, LIF].

Web Annotation was originally developed for adding shallow annotations (primarily labels, glosses or entity links) to web resources, but lacking the necessary data structures to represent complex data structures as needed for linguistic annotation. It is possible to complement Web Annotation with linguistic data structures [38], but these are not covered by the Web Annotation specification nor do they seem to be used in current practice.

The NLP Interchange Format (NIF) has been developed to facilitate the implementation of NLP workflows on the basis of web technologies. NIF is a representative of a broader class of RDF-based vocabularies designed for this purpose, e.g., TELIX [32], NAF-RDF [17], etc., but taken here as an example because it is relatively widely used and not tied to a specific piece of software. RDF-based NLP data models such as NIF provide linguistic data structures for a number of specific applications (part-of-speech tagging, entity linking, parsing, etc.), but they are extended according to the requirements of these applications.

CoNLL-RDF is grounded in the NIF vocabulary, but extends it in two important ways: (i) It introduces its own IRI fragment schema, based on a segmentation in sentences and tokens, and thus allows to refer to empty elements. (ii) For the representation of syntax and other, advanced levels of representation CoNLL-RDF complements NIF with POWLA data structures and is thus capable to represent *every* kind of linguistic annotation (a claim we inherit from the Linguistic Annotation Framework that POWLA formalizes [5]).

6 Summary and Outlook

CoNLL-RDF thus complements both Web Annotation and NIF with a model firmly grounded in state-of-the-art NLP *research* and used in mature NLP *applications*, and thus, better prepared for future applications based on current-day research. Within this paper, we provide the first formal account of the necessary data structures, and the first formalization of individual CoNLL dialects and related formats.

At the same time, however, CoNLL-RDF is not merely a data model, but it comes with a number of tools to facilitate the creation, manipulation and evaluation of linguistic annotations, as well as interoperability with "classical" formats in NLP and the language

[6] https://linguistic-lod.org/llod-cloud
[7] https://www.w3.org/2016/05/ontolex/

resource community. It is important here to note that CoNLL-RDF does not aim to replace these with an RDF substitute within NLP, but instead, it formalizes existing TSV formats, can be serialized in TSV and thus be seamlessly integrated in existing NLP workflows – for applications that benefit from graph data structures as opposed to tables (e.g., dependency syntax, coreference or semantic roles), applications that build on the integration of information from multiple, distributed sources or that are beyond the expressivity of an existing TSV format. The main application, however, is to establish interoperability among TSV formats and between these and knowledge graph technologies, and this is where we see the potential of CoNLL-RDF.

Despite the wide range of potential applications, we see CoNLL-RDF not as a potential replacement for either NIF or Web Annotation. Instead, it aims to provide a technological bridge between NLP standards and the RDF/Linked Data world. In the context of RDF vocabularies, we expect CoNLL-RDF to serve (along with Web Annotation, the NLP Interchange Format and ISO TC37 standards) as a main input for the development of a harmonized vocabulary for linguistic annotations on the web that is currently under development within the W3C Community Group Linked Data for Language Technology (LD4LT).[8]

The CoNLL-RDF ontology is available as part of the CoNLL-RDF repository and published under the same license (Apache 2.0).[9] The ACoLi CoNLL libraries (that contain CoNLL-RDF and CoNLL-Transform, along with other modules) are also open-source published and available from `https://github.com/acoli-repo/conll`. The ontology has been published under `http://purl.org/acoli/conll#`.

References

1 Frank Abromeit and Christian Chiarcos. Automatic Detection of Language and Annotation Model Information in CoNLL Corpora. In Maria Eskevich, Gerard de Melo, Christian Fäth, John P. McCrae, Paul Buitelaar, Christian Chiarcos, Bettina Klimek, and Milan Dojchinovski, editors, *2nd Conference on Language, Data and Knowledge (LDK 2019)*, volume 70 of *Open-Access Series in Informatics (OASIcs)*, pages 23:1–23:9, Dagstuhl, Germany, 2019. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

2 Alan Akbik, Laura Chiticariu, Marina Danilevsky, Yunyao Li, Shivakumar Vaithyanathan, and Huaiyu Zhu. Generating High Quality Proposition Banks for Multilingual Semantic Role Labeling. In *Proc. of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 397–407, 2015.

3 Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked Data – The Story So Far. *International Journal on Semantic Web and Information Systems*, 5(3):1–22, 2009.

4 Christian Chiarcos. A Generic Formalism to Represent Linguistic Corpora in RDF and OWL/DL. In *Proc. of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 3205–3212. ELRA, 2012.

5 Christian Chiarcos. POWLA: Modeling Linguistic Corpora in OWL/DL. In Elena Simperl, Philipp Cimiano, Axel Polleres, Oscar Corcho, and Valentina Presutti, editors, *The Semantic Web: Research and Applications*, pages 225–239, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[8] `https://www.w3.org/community/ld4lt/`, `https://github.com/ld4lt/linguistic-annotation`

[9] `https://doi.org/10.5281/zenodo.4361476/`

6 Christian Chiarcos, Kathrin Donandt, Hasmik Sargsian, M Ionov, and J Wichers Schreur. Towards llod-based language contact studies. a case study in interoperability. In *Proc. of the 6th Workshop on Linked Data in Linguistics (LDL)*, 2018.

7 Christian Chiarcos and Christian Fäth. CoNLL-RDF: Linked Corpora Done in an NLP-Friendly Way. In Jorge Gracia, Francis Bond, John P. McCrae, Paul Buitelaar, Christian Chiarcos, and Sebastian Hellmann, editors, *Language, Data, and Knowledge*, pages 74–88, Cham, Switzerland, 2017. Springer.

8 Christian Chiarcos and Christian Fäth. Graph-based annotation engineering: towards a gold corpus for role and reference grammar. In *2nd Conference on Language, Data and Knowledge (LDK 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2019.

9 Christian Chiarcos and Christian Fäth. Graph-Based Annotation Engineering: Towards a Gold Corpus for Role and Reference Grammar. In *2nd Conference on Language, Data and Knowledge (LDK-2019)*, pages 9:1–9:11. OpenAccess Series in Informatics, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, Germany, 2019.

10 Christian Chiarcos and Luis Glaser. A Tree Extension for CoNLL-RDF. In *Proc. of the Twelfth International Conference on Language Resources and Evaluation (LREC-2020)*, pages 7161–7169, Marseille, France, 2020. ELRA.

11 Christian Chiarcos, Ilya Khait, Émilie Pagé-Perron, Niko Schenk, Christian Fäth, Julius Steuer, William Mcgrath, Jinyan Wang, et al. Annotating a low-resource language with llod technology: Sumerian morphology and syntax. *Information*, 9(11):290, 2018.

12 Christian Chiarcos, Benjamin Kosmehl, Christian Fäth, and Maria Sukhareva. Analyzing Middle High German syntax with RDF and SPARQL. In *Proc. of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, pages 4525–4534, Miyazaki, Japan, 2018.

13 O. Christ. A modular and flexible architecture for an integrated corpus query system. In *Papers in Computational Lexicography (COMPLEX-1994)*, page 22–32, Budapest, Hungary, 1994.

14 Philipp Cimiano, Christian Chiarcos, John P. McCrae, and Jorge Gracia. *Linguistic Linked Data: Representation, Generation and Applications*. Springer International Publishing, Cham, 2020.

15 Souripriya Das, Seema Sundara, and Richard Cyganiak. R2RML: RDB to RDF Mapping Language. W3C Recommendation. `https://www.w3.org/TR/r2rml`, 2012.

16 Stefan Evert and Andrew Hardie. Twenty-first Century Corpus Workbench: Updating a Query Architecture for the New Millennium. In *Proc. of the Corpus Linguistics 2011 Conference*, pages 1–21, Birmingham, UK, 2011.

17 Antske Fokkens, Aitor Soroa, Zuhaitz Beloki, Niels Ockeloen, German Rigau, Willem Robert van Hage, and Piek Vossen. NAF and GAF: Linking Linguistic Annotations. In *Proc. of the Tenth Joint ISO-ACL SIGSEM Workshop on Interoperable Semantic Annotation*, pages 9–16, 2014.

18 Christian Fäth, Christian Chiarcos, Björn Ebbrecht, and Maxim Ionov. Fintan – Flexible, Integrated Transformation and annotation eNgineering. In *Proc. of the Twelfth International Conference on Language Resources and Evaluation (LREC-2020)*, pages 7212–7221, Marseille, France, 2020. ELRA.

19 A. Ghiran and R. A. Buchmann. Semantic Integration of Security Knowledge Sources. In *Twelfth International Conference on Research Challenges in Information Science (RCIS-2018)*, pages 1–9, 2018.

20 Noori Haider and Fokhray Hossain. CSV2RDF: Generating RDF Data from CSV File Using Semantic Web Technologies. *Journal of Theoretical and Applied Information Technology*, 96(20):6889–6902, 2018.

21 Sebastian Hellmann, Jens Lehmann, Sören Auer, and Martin Brümmer. Integrating NLP Using Linked Data. In Camille Salinesi, Moira C. Norrie, and Óscar Pastor, editors, *Advanced Information Systems Engineering*, volume 7908, pages 98–113. Springer Berlin Heidelberg, 2013.

22 Eero Antero Hyvönen, Petri Leskinen, Minna Tamper, and Jouni Antero Tuominen. Semantic National Biography of Finland. In Eetu Mäkelä, Mikko Tolonen, and Jouni Tuominen, editors, *Proc. of the DHN 2018*, CEUR Workshop Proceedings, pages 372–385, International, 2018. CEUR Workshop Proceedings.

23 N. Ide and L. Romary. International Standard for a Linguistic Annotation Framework. *Natural language engineering*, 10(3-4):211–225, 2004.

24 Maxim Ionov, Florian Stein, Sagar Sehgal, and Christian Chiarcos. cqp4rdf: Towards a suite for rdf-based corpus linguistics. In *European Semantic Web Conference*, pages 115–121. Springer, 2020.

25 ISO. Language Resource Management - Linguistic Annotation Framework (LAF). Standard, International Organization for Standardization, Geneva, 2012. Project leader: Nancy Ide.

26 Adam Kilgarriff, Vít Baisa, Jan Bušta, Miloš Jakubíček, Vojtěch Kovář, Jan Michelfeit, Pavel Rychlý, and Vít Suchomel. The Sketch Engine: Ten Years On. *Lexicography*, 1(1):7–36, 2014.

27 Christo Kirov, Ryan Cotterell, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sebastian J Mielke, Arya D McCarthy, Sandra Kübler, et al. UniMorph 2.0: Universal Morphology. In *Proc. of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, pages 1868–1873, 2018.

28 Francesco Mambrini and Marco Passarotti. Linked open treebanks. interlinking syntactically annotated corpora in the lila knowledge base of linguistic resources for latin. In *Proc. of the 18th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2019)*, pages 74–81, 2019.

29 Francesco Mambrini and Marco Passarotti. Linked Open Treebanks. Interlinking Syntactically Annotated Corpora in the LiLa Knowledge Base of Linguistic Resources for Latin. In *Proc. of TLT, SyntaxFest 2019*, pages 74–81, Paris, France, 2019. Association for Computational Linguistics.

30 Mitch Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993.

31 Joakim Nivre, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. Universal Dependencies v1: A Multilingual Treebank Collection. In *Proc. of the Tenth International Conference on Language Resources and Evaluation (LREC-2016)*, pages 1659–1666, 2016.

32 Emilio Rubiera, Luis Polo, Diego Berrueta, and Adil El Ghali. TELIX: An RDF-based Model for Linguistic Annotation. In *Extended Semantic Web Conference*, pages 195–209. Springer, 2012.

33 Robert Sanderson, Paolo Ciccarese, and Benjamin Young. Web Annotation Data Model. Technical report, W3C Recommendation, 2017. URL: `https://www.w3.org/TR/annotation-model/`.

34 Erik F Sang and Sabine Buchholz. Introduction to the CoNLL-2000 shared task: Chunking. *arXiv preprint*, 2000. `arXiv:cs/0009008`.

35 Helmut Schmid. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proc. of International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK, 1994.

36 Minna Tamper, Petri Leskinen, Kasper Apajalahti, and Eero Hyvönen. Using biographical texts as linked data for prosopographical research and applications. In *Euro-Mediterranean Conference*, pages 125–137. Springer, 2018.

37 Marc Verhagen, Keith Suderman, Di Wang, Nancy Ide, Chunqi Shi, Jonathan Wright, and James Pustejovsky. The LAPPS Interchange Format. In *International Workshop on Worldwide Language Service Infrastructure*, pages 33–47. Springer, 2015.

38 Karin Verspoor and Kevin Livingston. Towards Adaptation of Linguistic Annotations to Scholarly Annotation Formalisms on the Semantic Web. In *Proc. of the Sixth Linguistic Annotation Workshop*, pages 75–84, 2012.

On the Utility of Word Embeddings for Enriching OpenWordNet-PT

Hugo Gonçalo Oliveira
CISUC, Department of Informatics Engineering, University of Coimbra, Portugal

Fredson Silva de Souza Aguiar
FGV/EMAp, Rio de Janeiro, Brazil

Alexandre Rademaker
IBM Research, Rio de Janeiro, Brazil
FGV/EMAp, Rio de Janeiro, Brazil

Abstract

The maintenance of wordnets and lexical knwoledge bases typically relies on time-consuming manual effort. In order to minimise this issue, we propose the exploitation of models of distributional semantics, namely word embeddings learned from corpora, in the automatic identification of relation instances missing in a wordnet. Analogy-solving methods are first used for learning a set of relations from analogy tests focused on each relation. Despite their low accuracy, we noted that a portion of the top-given answers are good suggestions of relation instances that could be included in the wordnet. This procedure is applied to the enrichment of OpenWordNet-PT, a public Portuguese wordnet. Relations are learned from data acquired from this resource, and illustrative examples are provided. Results are promising for accelerating the identification of missing relation instances, as we estimate that about 17% of the potential suggestions are good, a proportion that almost doubles if some are automatically invalidated.

2012 ACM Subject Classification Computing methodologies → Lexical semantics; Computing methodologies → Language resources

Keywords and phrases word embeddings, lexical resources, wordnet, analogy tests

Digital Object Identifier 10.4230/OASIcs.LDK.2021.21

Funding *Hugo Gonçalo Oliveira*: Supported by national funds through FCT, within the scope of the project CISUC (UID/CEC/00326/2020) and by European Social Fund, through the Regional Operational Program Centro 2020.

Acknowledgements The research described in this paper was partially conducted in the scope of the COST Action CA18209 Nexus Linguarum (European network for Web-centred linguistic data science).

1 Introduction

When it comes to representing lexico-semantic knowledge, there are two main approaches: lexical knowledge bases, like wordnets [13], and distributional models, like word embeddings [24] learned from raw text. Wordnets are more formalised than distributional models, and typically rely on some manual effort, often by experts, e.g., for grouping synonymous words in so-called synsets and linking them according to a small set of semantic relations with lexicographic relevance, such as hypernymy and meronymy. On the other hand, distributional models are inspired by the distributional hypothesis [19] and capture the meaning of the words of a language by analysing their neighbourhoods in large collections of text.

Even though they are not formalised at all, word embeddings can be learned automatically and do not require expert knowledge. Moreover, from the regularities in natural language text, they may capture virtually any semantic relation between words, even if not all can be acquired with simple methods, such as the vector offset [24]. This suggests that word embeddings can be of great value for minimising some of the limitations of wordnets, namely their coverage of relation instances.

In this paper, we explore Portuguese word embeddings having in mind the enrichment of OpenWordnet-PT (OWN-PT) [8], a public domain Portuguese wordnet in the Open Multilingual WordNet (OMW) [1] project, aligned with Princeton WordNet (PWN) [13], and with a comprehensive coverage of the language. More precisely, we: (i) create several analogy tests with data extracted from OWN-PT, each for a different relation; (ii) apply two analogy-solving methods [11] to the previous test, though with poor performance; (iii) inspect the top answers given by one of the methods and conclude that some correspond to missing relation instances in OWN-PT, which can thus be used as suggestions for its enrichment.

The remaining of the paper is organised as follows: in Section 2, we overview work on the automatic acquisition of lexico-semantic relations from text, their usage for enriching wordnets, as well as some examples of how word embeddings can be exploited for this purpose, using analogy-solving methods; in Section 3, we give a general overview of OWN-PT; in Section 4, we describe the applied methods and how we created analogy tests with OWN-PT, further used for learning how relations are represented in word embeddings; in Section 5, we report on the accuracy of analogy-solving methods; in Section 6, following an inspection of the answers given by the previous methods, we discuss on the utility of such methods for enriching OWN-PT; in Section 7, we highlight the main conclusions of this work.

2 Background and Related Work

Earlier attempts for the automatic acquisition of lexico-semantic relations and their compilation in a lexical knowledge base exploited language dictionaries, their structure, and patterns used in the definitions [5]. Once Princeton WordNet (PWN) [13] became available for English, work on the creation of such a resource from scratch was no longer a priority.

Following the success of PWN, wordnets were developed for many other languages [2]. However, PWN is the product of intensive manual labour during many years. So, the creation of wordnets varied from project to project. Roughly, two approaches have been followed for creating wordnets [35]: the *expand approach* translates the synsets in PWN to a target language, takes over the relations from PWN, and revises them; the *merge approach* defines synsets and relations in a language and then aligns them with PWN, using equivalence relations. Instead of starting from scratch with the merge approach, the expand approach is the most commonly used among wordnets in the Open Multilingual WordNet initiative.[1]

But the truth is that, no matter the approach taken, fixes will always be required in a wordnet, and having an adequate coverage will always be an issue. Not to mention that language keeps evolving and maintenance is always necessary. Therefore, it is no surprise that different automatic procedures have been proposed for enriching wordnets, most of which by exploiting raw textual corpora. Such work ranges from handcrafting useful patterns for acquiring hypernymy-hyponymy relations [21], to learning similar patterns, not only for hypernymy [32], but also other relations [28], following weakly-supervised approaches that used examples from PWN as seeds.

In the last decade, more efficient distributional representations of words became available [24, 29], with promising results regarding lexical tasks, like computing word similarity and analogies. The former aims at computing the similarity between pairs of words, e.g. *dog* should be more similar to *cat* than to *car*. Performance is typically assessed with tests where similarity was manually assigned to pairs of words.

[1] `http://compling.hss.ntu.edu.sg/omw/`

Computing an analogy consists of answering the question *what is to b as a^* is to a?*, e.g. *what is to Portugal as Paris is to France?*. In this case, the relation between the computed word, b^*, and b, must be as close as possible to the relation between a^* and a. But the number of possible relations between two words is huge, especially if we consider morphological and semantic, and different relations will pose different challenges. Therefore, analogy tests, used for assessing this task, typically cover different relation types. For instance, the Google Analogy Test (GAT), notably used for assessing word2vec embeddings [24], covers nine types of syntactic and five types of semantic relation. The Bigger Analogy Test Set (BATS) [15] covers a total of 40 relation types, 10 for each of four categories: grammatical inflections, word-formation, lexicographic and world-knowledge relations.

The most common method for computing an analogy in the embedding space is to compute the vector offset, also known as the 3CosAdd method [24]. Yet, alternative methods were proposed for minimising limitations of the previous method. For instance, in addition to releasing BATS, its creators propose two methods that, instead of computing an analogy from a single pair (a and a^*), consider a set of vectors between pairs of words related the same way [11]. To some extent, these methods, baptised as 3CosAvg and LRCos, can generalise the vectors that represent the target relation, and thus be used for relation discovery.

3CosAvg and LRCos have shown to perform better, not only for English [11], but also for Portuguese, where they have been used for solving a translation of GAT [33] and also a newly created dataset, TALES, focused on Portuguese lexico-semantic relations [17]. The latter work also showed that lexico-semantic analogies are significantly more challenging to solve, because there are many relation instances sharing the same argument, thus allowing for several correct answers. In fact, sometimes, correct answers are just too many to be included in a dataset or lexical resource. This further suggests that these methods can be useful for automatically suggesting potentially missing links in a lexical resource.

The aforementioned distributional representations lately became known as static word embeddings, because they have a single representation for each word, while neural language models, like BERT [10], are based on contextual embeddings, i.e., the same word is represented differently, depending on its context. There is recent work on using neural language models in related tasks, such as filling blanks in short sentences that denote specific semantic relations, and thus discovering relations of such types [30, 3]; word sense disambiguation [36], given their contextual representations; and even analogy-solving [12], despite the lack of context in analogy tests. However, exploring those models is out of the scope of this work.

Soon, researchers noted that analogy-solving methods could be assessed in the discovery of morphological and semantic relations, including lexico-semantic, from word embeddings [15]. Moreover, other researchers assumedly used word embeddings for extending wordnets, e.g. for discovering new synsets and scoring candidate hypernyms by combining distances in the wordnet graph and their distributional similarity [31]. Others worked on the automatic construction of the whole wordnet from scratch, using word embeddings, in addition to bilingual dictionaries [22].

Wordnets, focused on lexical knowledge, were also extended with world knowledge, e.g., by linking them with Wikipedia, as in the BabelNet project [25]. For Portuguese, on this scope, Onto.PT is an automatically-created wordnet [16] that combines information in existing thesauri with relations extracted from several Portuguese dictionaries [18]. On the other hand, OpenWordNet-PT [8], used in this work, is a Portuguese wordnet aligned with PWN, originally developed as a syntactic projection of the Universal WordNet [7], but, since then, manually maintained (see more in Section 3).

3 OpenWordNet-PT

OWN-PT is an ongoing project to create a large wordnet for Portuguese. It has currently 52,559 synsets, 52,210 word forms and 83,841 senses.[2] It is the Portuguese wordnet in the Open Multilingual WordNet (OMW) [1] project, Freeling [26], BabelNet [25] and Google Translate.[3] OWN-PT synsets are aligned with the corresponding PWN synset and relations among the PWN synsets are projected to the OWN-PT synsets. OWN-PT is distributed in RDF following the vocabulary first described by de Paiva et al. [9].

In PWN, the main relation among words is synonymy. Synonyms – words that denote the same concept and are interchangeable in many contexts – are grouped into synsets. Each PWN synset is linked to other synsets by means of a small number of conceptual relations. Word forms[4] with several distinct meanings are represented in as many distinct synsets. Thus, each form-meaning pair (i.e., a word sense, the occurrence of a word in a synset) in PWN is unique. Synsets and word senses are interlinked by means of conceptual-semantic and lexical relations. The latter hold between word senses, whereas semantic relations hold between synsets; and there is also a small set of relations between synsets and word senses. Examples of semantic relations in PWN are: hyperonym, hyponym, meronym/holonym (part, substance and member), troponyms. Examples of lexical relations are: antonym and derivationally related.

The majority of the PWN relations connect words of the same part-of-speech (POS). Thus, PWN really consists of four sub-networks, respectively for nouns, verbs, adjectives and adverbs, with few cross-POS pointers. Cross-POS relations include the "morphosemantic" links that hold among semantically similar words sharing a stem with the same meaning, e.g., observe (verb), observant (adjective), observation and observatory (nouns). In many of the noun-verb pairs (i.e., nominalizations) the semantic role of the noun with respect to the verb has been specified, e.g., "painter" is the agent of "paint" (verb) while "painting" and "picture" is its result.

OWN-PT synsets are also classified into two additional classes: Core and Base. "Core" synsets are obtained from a semi-automatically compiled list with the 5,000 most frequently used word senses, followed by some manual filtering and adjustment by the PWN team [4]. The notion of base concepts was introduced in the EuroWordNet project [35] to reach maximum overlap and compatibility across wordnets in different languages. At the same time, this allows for the distributive development of wordnets in the world, each wordnet being a language specific structure and lexicalization pattern. "Base" Concepts are selected to be those that play an important role in the various wordnets of different languages.

4 Analogy Tests from OpenWordNet-PT Contents

Our main goal was to explore static word embeddings in the discovery of relation instances that could be useful for enriching OWN-PT. We thus needed an implementation of useful methods for this purpose, as well as data for training and assessing them.

The most common method for computing an analogy in the embedding space is to compute the vector offset [24], also known as the 3CosAdd method (Equation 1).

$$b^* = \underset{w \in V}{\operatorname{argmax}} \cos(w, a^* - a + b) \tag{1}$$

[2] Numbers can be compared to other open wordnets listed in the OMW at `http://compling.hss.ntu.edu.sg/omw/`.
[3] `https://translate.google.com/intl/en/about/license/`
[4] The term "word form" refers to single words or multi-word expressions.

Yet, as referred in Section 2, analogy-solving methods like 3CosAvg and LRCos suit our purpose better, because they exploit such embeddings for learning the relation between several pairs of words. 3CosAvg (Equation 2) computes the average offset between words in position a and respective words in position a^*, in a set of relation instances of the target type. The answer, b^*, must maximise the cosine with the vector resulting from summing the average offset to b.

$$b^* = \underset{w \in V}{\operatorname{argmax}} \cos(w, b + avg\_offset) \tag{2}$$

LRCos (Equation 3) considers the probability that a word w belongs to the same class as other words in position a^*, as well as the similarity between w and b, measured with the cosine. Although any classification algorithm could be used for this, the default implementation of LRCos relies on logistic regression for computing the likelihood of a word belonging to the class of words a^*.

$$b^* = \underset{w \in V}{\operatorname{argmax}} P(w \in target\_class) * cos(w, b) \tag{3}$$

In order to analyse how well the previous methods could learn a selection of relations in OWN-PT, we adopted Vecto,[5] a package for loading static word embeddings that includes implementations of 3CosAvg and LRCos, and supports analogy tests in the format of the BATS test [15]. For this purpose, analogy tests were created from OWN-PT. Table 1 presents the twelve relations considered in their production. This choice was guided by the number of instances available (see below), but also by the kind of relations that we believe could be learned from word embeddings. Therefore, relations like "see also", "classified by" and "same verb group" were discarded.

Analogy tests are organized in two-column tabular text files. Each test has several lines with a question word, in the first column; and a list of possible answers, in the second. All the words in the answer have to be related to the question word, according to OWN-PT. A different test was created for each relation, meaning that, in the same test, the relation between the question words and those in the answer was always the same. Figure 1 illustrates the format of the analogy test files with examples for three relations. For better understanding, rough translations were added for each line, but they are not part of the test.

For the creation of the tests, each conceptual-semantic relation instance between synsets was first expanded into a cartesian product of their word senses, using the SPARQL query in Listing 1. This query can be submitted to the OWN-PT SPARQL endpoint at `http://openwordnet-pt.org`.

Then, we group the instance pairs for each relation by their first projection (source, first column) and list all the related words in the second column (target, second column). Several experiments were made, for further improving the quality of the tests, given our goal. For instance, it is expected that the analogy-solving methods will learn better representations from single-sense words that are frequent enough in corpora. Specifically, in this work, we decided to consider only lines where the question word is in a "Core" synset. Moreover, the words in the answer were ordered so that words in "Core" synsets, if there were any, and words with fewer senses were listed first. This became relevant once we noticed that, in the training phase, Vecto considers only the first word in the list of possible answers. After this, we decided not to use tests with fewer than 30 questions (lines), or with more than 1,000,

[5] `https://github.com/vecto-ai`

atividade (activity)	ativo/agencioso/inativo (active / inactive)
bem-estar (wellness)	doentio/adoentado/insalubre/salubre/doente/são/saudável (sick/diseased/salubrious/unhealthy/healthy)
bondade (goodness)	boa/bom (good)
aberto (open)	fechado (closed)
abstrato (abstract)	concreto (concrete)
alto (tall)	baixo (low)
pálpebra (eyelid)	celha/conjuntiva/pestana/cílio (lash/conjunctiva/eyelash/cilium)
pássaro (bird)	plumagem/ala/bico/pluma/asa/pena/culatra/fúrcula/garupa/... (plumage/wing/beak/feather/wing/rump/croup)
pé (foot)	calcanhar/alfândega/artelho/polegada/dedo/calcâneo/hálux/sola (heel/calf/ankle/insole/finger/sole)

Figure 1 Excerpts of the generated datasets and rough translations, for the relations: attribute, antonymOf, partMeronymOf.

Listing 1 SPARQL query to produce the input data for Vecto.

```
select ?au ?t1 ?rel ?t2 (group_concat(?bu; separator = "/") AS ?values)
{
 ?s1 wn30:containsWordSense ?ss1 ; a ?t1 ; a wn30:CoreSynset .
 ?s1 skos:inScheme <http://logics.emap.fgv.br/wn/> .
 ?ss1 wn30:word/wn30:lexicalForm ?a .
 ?s2 wn30:containsWordSense ?ss2 ; a ?t2 .
 ?ss2 wn30:word/wn30:lexicalForm ?b .
 ?s1 skos:inScheme <http://logics.emap.fgv.br/wn/> .
 BIND(replace(lcase(str(?b))," ","_") AS ?bu)
 BIND(replace(lcase(str(?a))," ","_") AS ?au)

 ?s1en ?rel ?s2en .
 ?s1en owl:sameAs ?s1 .
 ?s2en owl:sameAs ?s2 .
}
group by ?au ?rel ?t1 ?t2
```

which included, for instance, hypernymOf and hyponymOf. If few questions would not be enough for generalizing the relations, the option for not considering larger tests was mostly practical, having in mind the manual validation and analysis of the results. This does not mean that, in the future, these relations cannot be considered as well.

5 Accuracy in Relation Learning

In order to run 3CosAvg and LRCos in the OWN-PT analogy tests, we used Vecto on the 300-sized Portuguese GloVe embeddings from the NILC repository [20]. This choice was supported by previous works, for English [11] and for Portuguese [33, 17], where GloVe embeddings have shown to perform better when it comes to solving semantic analogies.

At a lower level, each analogy-solving method is trained with every line of the test – corresponding to the question word (first column) and the answer (first word in the second column) – except one, and then tested on the remaining line, i.e., given the word in the target question (b), the model learned from all other questions and their answers tries to predict one of the words in its answer (b^*). In the end, Vecto computes the average accuracy of repeating the previous process for every question in the test.

For every considered relation, Table 1 shows the number of questions in its test and the accuracies achieved with the analogy solving methods – 3CosAvg and LRCos – and also with simple similarity (SimToB), here used as a baseline. For each question word, the latter consists of answering with its most similar word in the embeddings, i.e., the one maximising the cosine similarity. This also helps to take conclusions on whether the analogy-solving methods are improving upon this simple computation.

In fact, for eight out of 12 relations, analogy-solving methods lead to improvements, and there is only one (memberMeronymOf) for which both of them perform below the baseline. For the former eight relations, the best performance is achieved with LRCos, whereas for three of the remaining four 3CosAvg matches the performance of the baseline. Out of them, the accuracy of LRCos is 0 for the relation for which available data is less (substanceHolonym).

Table 1 Accuracy of the 3CosAvg and LRCos methods in different types of relation.

		Accuracy		
Relation	**Questions**	**SimToB**	**3CosAvg**	**LRCos**
agent	75	1.3%	1.3%	**29.3%**
antonymOf	68	**19.1%**	**19.1%**	7.4%
attribute	88	2.3%	9.1%	**21.6%**
byMeansOf	41	14.6%	26.8%	**46.3%**
causes	60	5.0%	6.7%	**8.3%**
entails	123	5.7%	5.7%	**6.5%**
memberHolonymOf	157	**5.1%**	**5.1%**	4.5%
memberMeronymOf	77	**10.4%**	7.8%	3.9%
partHolonymOf	417	2.2%	3.1%	**7.2%**
partMeronymOf	569	1.2%	1.2%	**3.0%**
substanceHolonymOf	33	**6.1%**	**6.1%**	0.0%
substanceMeronymOf	84	1.2%	2.4%	**7.1%**

Still, despite the noted improvements over the baselines, accuracies are still poor – for LRCos, only three (agent, attribute, byMeansOf) are above 20% and none is above 50%. The more homogeneous the first arguments of a relation are, the better LRCos seems to perform, which makes sense, because it makes the task of the classifier easier. For instance, in the byMeansOf and agent relations, first arguments are of a specific kind of verb, which favors the underlying classification, considered by LRCos. Despite some improvements, the performance of 3CosAvg is more in line with the baseline, with higher accuracy for relations with more semantically-similar arguments, starting with antonymOf.

On the one hand, figures show that generalising lexico-semantic relations in word embeddings is a challenging task, even if much more challenging for some relations (e.g., part) than for others (e.g., attribute, byMeansOf). On the other hand, accuracy is computed in OWN-PT, a resource that tries to cover the whole Portuguese language but is in constant development and, as it happens for all wordnets, has its gaps.

Moreover, accuracy is far from telling the whole story, because it only considers the first answer. Despite this fact, the report generated by Vecto also provides the top-n answers for each question. And if, for some relations typically found in an analogy test (e.g., morphological relations, country-capital, country-currency) questions tend to have a single answer, this does not happen for many lexico-semantic relations, e.g., an object will generally have several parts, and an attribute will have several possible values. Following the aforementioned reasons, we saw the list of top answers given as a useful source of suggestions for new relation instances in OWN-PT, i.e., the approach taken could be seen as an automatic way of providing such suggestions.

To better illustrate this, we show the top-8 answers for "*aperfeiçoar* (ameliorate) causes *b**", after automatic lemmatization (see Section 6) and removal of resulting duplicates: *aprender* (learn), *rever* (review), *trabalhar* (work), *repensar* (rethink), *evoluir* (evolve), *precisar* (need), *progredir* (progress), *melhorar* (improve). Out of them, only one is in OWN-PT (*melhorar*), in the eighth position, while six others could be considered as correct, but are just not in OWN-PT. Of course that there are also questions with no useful answers like, for instance, "sagrado (sacred) antonymOf *b**". Out of the answers for this question, only three matched the adjective POS: *eterno* (eternal), *religioso* (religious) and *obscuro* (obscure). Even if a different relation could possibly be established between some of them, none is an antonym of the question word. Next section tries to better quantify the proportion of potentially useful relations that could be suggested by this approach, with a manual validation.

6 Utility Analysis

Following the experiment reported in the previous section and the considerations regarding the potential utility of the given answers, we aimed at better quantifying that utility. This was necessary for better ascertaining the applicability of the analogy-solving methods for enriching wordnets, specifically OWN-PT.

For this purpose, we sampled a list of relation instances for manual inspection and human validation. As a preliminary validation, the criteria for selecting the relations to sample were pragmatic: we tried to cover four significantly different relation types, with varying performances in the first experiment (Section 5), also having in mind how easy it would be for a human to judge on their quality. Such a selection would mean a conservative estimation of the benefits of the proposed approach for enriching OWN-PT. It would also confirm the limited conclusions one can take from the accuracy values achieved and the preliminary inspection of the given answers.

For each selected relation, the sample included ten questions and their answers by LRCos, the method with the highest accuracy for more relations. Validation consisted of judging whether a relation of the given type actually holds between the question and each of the answers (e.g., *dente* parHolonymOf *cabeça*?).

Evaluating semantic relations between out-of-context words is always a challenging task. Despite this fact, as a preliminary evaluation, we decided to keep it simple and our main focus was on judging whether a relation of the target type can actually hold between the question and each of the answers (e.g., *largura* (width) attribute *transversal* (transversal)?). The sample was to be annotated in a spreadsheet, with relations meaning clarified by canonical examples (e.g., attribute *altura*-NOUN → *alto*-ADJ, in English, height-NOUN, high-ADJ). A human annotator had to label the suggested relation as Correct (i.e., the relation may hold for the question-answer pair) or Incorrect (i.e., the relation does not hold for the pair).

Yet, in order to accelerate human validation, some answers in the sample were automatically validated before presented to the annotators. This was performed with the help of MorphoBR [6], a large-coverage full-form lexicon for the morphological analysis of Portuguese, and included the following checks:

- If the POS of the answer did not match the POS of the range of the target relation, it was automatically labelled as invalid. For instance, if a relation is defined to hold between nouns and adjectives (e.g., attribute), answers that were not found in the lexicon's adjectives would fail this test;
- If the POS of the answer matched the POS of the range of the target relation but was not in the lemma form, the answer was lemmatized. In a minority of cases, this could lead to duplicate answers.

Moreover, if the answer was already in OWN-PT, it was automatically labeled as correct.

Table 2 summarises the results of this manual validation when made by one of the authors of this paper, who is part of the team that maintains OWN-PT. It organises the answers into those: corresponding to relation instances already in OWN-PT; invalid due to incompatible POS; or not in OWN-PT, but labelled as Correct. For some instances, the annotator provided an additional comment that the relation is incorrect, but a relation of a different type indeed holds between question and answer (e.g., synonymy instead of antonymy).

In a later stage, the sample was also validated by another author of the paper, which enabled us to measure the Cohen's Kappa κ. When the automatically labeled entries are not considered, κ was 0.63, which corresponds to substantial agreement [23].

Table 2 Summary of manual validation of 376 pairs of words, covering four relations, by one human annotator (OWN-PT maintainer). Numbers in parenthesis are percentages for each relation, considering all the entries in the sample.

Relation	Total	In OWN-PT	Invalid	Correct	Other Relation
antonymOf	90	0	39 (43%)	3 (3%)	26 (29%)
attribute	94	3 (3%)	35 (37%)	27 (29%)	23 (24%)
causes	95	2 (2%)	39 (41%)	10 (11%)	4 (4%)
partHolonym	97	6 (6%)	22 (23%)	26 (27%)	17 (18%)

We see that, depending on the relation, useful suggestions vary significantly. For instance, for antonymyOf only three were labeled as correct, whereas for attribute and partHolonym more than a quarter of the suggestions were good, respectively 29% and 27%. It is also clear that these figures are not proportional to the accuracies achieved for each relation in Section 5, confirming that those results are of limited application. For instance, the accuracy for the aforementioned relations with LRCos is as different as 21% (attribute) and 7% (partHolonymOf).

Despite the simplicity of the task, some non-trivial examples were not hard to find. For instance, *ponta* (lead, end, point, or tip) is indeed a part holonym of many objects (i.e., many objects do have a tip), but the challenge is to identify those objects where it is important to have this relation explicit. Among the good findings, some could be added to OWN-PT right away, including the following examples:

- *integrado* (integrated) antonym of *separado* (separate);
- *ideologia* (ideology) attribute *marxista* (Marxist);
- *aperfeiçoar* (ameliorate) causes *evoluir* (evolve);
- *dente* (tooth) part holonym of *elefante* (elephant);

Considering all four relations in the sample, the proportion of useful suggestions is about 17%. Yet, we should note that a significant proportion (39%) of the suggestions was automatically labeled, most of which for being invalid. This made it possible to decrease the amount of suggestions that required human validation. If such suggestions are ignored, the proportion of useful suggestions is close to 29%. This shows the potential of the proposed approach for accelerating the process of enriching wordnets, by suggesting the inclusion of relation instances that are missing from the resource. At the same time, this proportion confirms that the process needs human intervention, i.e., we cannot simply add all suggestions automatically. In fact, a second step is still required for selecting the attachment points

in OWN-PT, i.e., the synsets corresponding to the arguments of the suggested relations. Furthermore, the example of *ponta* suggests that better inclusion criteria are needed to improve human judgment.[6]

7 Conclusion

This paper described how methods for automatic analogy-solving with word embeddings were applied to the discovery of lexico-semantic relations in Portuguese. It further analysed the utility of discovered relation instances for enriching OWN-PT, a Portuguese wordnet. Even if the accuracy of such methods is poor, among other challenges, it is harmed by the gaps in the wordnet, resulting in the consideration of some answers that would be correct, as incorrect. Yet, as we have shown, some of the given answers are good suggestions for manual inclusion in the wordnet. In a small validated sample of answers, we found about 17% good suggestions. We also noted that some suggestions can be automatically labeled as invalid, leading to about 29% suggestions out of all that required human validation. We thus see the described approach as a promising avenue for finding gaps and enriching wordnets. Although applied to Portuguese, a similar procedure could be adopted for other languages for which a wordnet and a model of word embeddings are available. Still, this was just a preliminary validation. An evaluation considering more answers and all relation types should be performed in the future. Such an exercise may also enable an analysis of the confusion between relations, and possibly identify actual errors in OWN-PT.

Despite accelerating the process, human intervention is always required for discriminating correct suggestions. Moreover, since this approach is based on word representations and not word senses, a human will also be necessary to find the suitable attachment points (i.e., word senses) for the suggested relation instance in the wordnet. So far, when a relation instance involved a lemma not covered by the wordnet, this lemma was added to a proper synset, if there was one. If not, nothing was done. In the future, this might lead to the creation of new synsets.

The process of enriching and maintaining a wordnet is never over, and so is not this work. In the near future, we aim to make the process of relation suggestion from word embeddings more flexible. In addition to lemmatization and exclusion criteria (i.e., valid POS) already applied to the obtained suggestions, we will work on isolating the analogy-solving methods from Vecto, which will enable to select only a controlled subset of relations for training, and then apply the learned models to a broader test set. A controlled training set could consider only core concepts or single-sense words, and possibly also features like word frequency, concreteness / imageability [27], experiential familiarity [14], among others. At the same time, a different test set will enable the discovery of relations for any word.

It is also our intention to explore neural language models for this process. As others have shown [30, 3], BERT's masked language model can be used as source of relational knowledge. We could probably adopt their approaches for Portuguese, using a BERT model pretrained for our language [34]. Finally, it would be interesting to consider word senses in the process. This could be explored in the discovery step and include the exploitation of contextual embeddings, e.g., from BERT; or in the validation step, where looking at the discovered relations in context, ideally with disambiguated words, should help the human judgement.

[6] In `https://globalwordnet.github.io/gwadoc/` there is an initial attempt at consistent documentation and examples for semantic/lexical relations used by different wordnets.

References

1 Francis Bond and Ryan Foster. Linking and extending an open multilingual wordnet. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1352–1362, 2013.

2 Francis Bond and Kyonghee Paik. A survey of wordnets and their licenses. *Small*, 8(4):5, 2012.

3 Zied Bouraoui, Jose Camacho-Collados, and Steven Schockaert. Inducing relational knowledge from bert. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7456–7463. AAAI, 2020.

4 Jordan Boyd-Graber, Christiane Fellbaum, Daniel Osherson, and Robert Schapire. Adding dense, weighted connections to wordnet. In *Proceedings of the third international WordNet conference*, pages 29–36. Citeseer, 2006.

5 Nicoletta Calzolari, Laura Pecchia, and Antonio Zampolli. Working on the italian machine dictionary: a semantic approach. In *COLING 1973 Volume 2: Computational And Mathematical Linguistics: Proceedings of the International Conference on Computational Linguistics*. Association for Computational Linguistics, 1973.

6 Leonel Figueiredo de Alencar, Bruno Cuconato, and Alexandre Rademaker. Morphobr: An open source large-coverage full-form lexicon for morphological analysis of portuguese. *Texto Livre: Linguagem e Tecnologia*, 11(3):1–25, 2018.

7 Gerard De Melo and Gerhard Weikum. Towards a universal wordnet by learning from combined evidence. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 513–522, 2009.

8 Valeria de Paiva, Alexandre Rademaker, and Gerard de Melo. OpenWordNet-PT: An Open Brazilian WordNet for Reasoning. In *Proceedings of 24th International Conference on Computational Linguistics*, COLING (Demo Paper), 2012.

9 Valeria de Paiva, Livy Real, Alexandre Rademaker, and Gerard de Melo. Nomlex-pt: A lexicon of portuguese nominalizations. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, Reykjavik, Iceland, 2014. European Language Resources Association (ELRA).

10 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, NAACL-HLT 2019, pages 4171–4186. Association for Computational Linguistics, 2019.

11 Aleksandr Drozd, Anna Gladkova, and Satoshi Matsuoka. Word embeddings, analogies, and machine learning: Beyond king - man + woman = queen. In *Proceedings the 26th International Conference on Computational Linguistics: Technical papers COLING 2016*, COLING 2016, pages 3519–3530, 2016.

12 Kawin Ethayarajh. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, 2019.

13 Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press, 1998.

14 Morton A Gernsbacher. Resolving 20 years of inconsistent interactions between lexical familiarity and orthography, concreteness, and polysemy. *Journal of experimental psychology: General*, 113(2):256, 1984.

15 Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuoka. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the NAACL 2016 Student Research Workshop*, pages 8–15. ACL, 2016.

16 Hugo Gonçalo Oliveira and Paulo Gomes. ECO and Onto.PT: A flexible approach for creating a Portuguese wordnet automatically. *Language Resources and Evaluation*, 48(2):373–393, 2014.

17 Hugo Gonçalo Oliveira, Tiago Sousa, and Ana Alves. TALES: Test set of Portuguese lexical-semantic relations for assessing word embeddings. In *Proceedings of the ECAI 2020 Workshop on Hybrid Intelligence for Natural Language Processing Tasks (HI4NLP 2020)*, volume 2693 of *CEUR Workshop Proceedings*, pages 41–47. CEUR-WS.org, 2020.

18 Hugo Gonçalo Oliveira, Diana Santos, Paulo Gomes, and Nuno Seco. PAPEL: A dictionary-based lexical ontology for Portuguese. In *Proceedings of Computational Processing of the Portuguese Language - 8th International Conference (PROPOR 2008)*, volume 5190 of *LNCS/LNAI*, pages 31–40, Aveiro, Portugal, September 2008. Springer.

19 Zelig Harris. Distributional structure. *Word*, 10(2-3):1456–1162, 1954.

20 Nathan S. Hartmann, Erick R. Fonseca, Christopher D. Shulby, Marcos V. Treviso, Jéssica S. Rodrigues, and Sandra M. Aluísio. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. In *Proceedings 11th Brazilian Symposium in Information and Human Language Technology (STIL 2017)*, 2017.

21 Marti A. Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of 14th Conference on Computational Linguistics*, COLING 92, pages 539–545, Morristown, NJ, USA, 1992. Association for Computational Linguistics.

22 Mikhail Khodak, Andrej Risteski, Christiane Fellbaum, and Sanjeev Arora. Automated wordnet construction using word embeddings. In *Proceedings of the 1st Workshop on Sense, Concept and Entity Representations and their Applications*, pages 12–23, 2017.

23 J Richard Landis and Gary G Koch. The measurement of observer agreement for categorical data. *Biometrics*, pages 159–174, 1977.

24 Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *Proceedings of the Workshop track of ICLR*, 2013.

25 Roberto Navigli and Simone Paolo Ponzetto. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250, 2012.

26 Lluís Padró and Evgeny Stanilovsky. FreeLing 3.0: Towards wider multilinguality. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2473–2479, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA).

27 Allan Paivio, John C Yuille, and Stephen A Madigan. Concreteness, imagery, and meaningfulness values for 925 nouns. *Journal of experimental psychology*, 76(1p2):1, 1968.

28 Patrick Pantel and Marco Pennacchiotti. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Procs of 21st International Conference on Computational Linguistics and 44th annual meeting of the Association for Computational Linguistics*, pages 113–120, Sydney, Australia, 2006. ACL Press.

29 Jeffrey Pennington, Richard Socher, and Christopher D. Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2014, pages 1532–1543. ACL, 2014.

30 Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473. Association for Computational Linguistics, 2019.

31 Heidi Sand, Erik Velldal, and Lilja Øvrelid. Wordnet extension via word embeddings: Experiments on the norwegian wordnet. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 298–302, 2017.

32 Rion Snow, Daniel Jurafsky, and Andrew Ng. Learning syntactic patterns for automatic hypernym discovery. *Advances in neural information processing systems*, 17:1297–1304, 2005.

33 Tiago Sousa, Hugo Gonçalo Oliveira, and Ana Alves. Exploring different methods for solving analogies with Portuguese word embeddings. In *Proceedings 9th Symposium on Languages, Applications and Technologies, SLATE 2020, July 13-14, 2020, School of Technology, Polytechnic Institute of Cávado and Ave, Portugal*, volume 83 of *OASIcs*, pages 9:1–9:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.

34 Fábio Souza, Rodrigo Nogueira, and Roberto Lotufo. Bertimbau: Pretrained bert models for brazilian portuguese. In *Proceedings of the Brazilian Conference on Intelligent Systems (BRACIS 2020)*, volume 12319 of *LNCS*, pages 403–417, Cham, 2020. Springer.

35 P. Vossen. *EuroWordNet: A multilingual database with lexical semantic networks*. Computers and the humanities. Springer Netherlands, 1998.

36 Gregor Wiedemann, Steffen Remus, Avi Chawla, and Chris Biemann. Does bert make any sense? interpretable word sense disambiguation with contextualized embeddings. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019): Long Papers*, pages 161–170, Erlangen, Germany, 2019. German Society for Computational Linguistics & Language Technology.

Towards Learning Terminological Concept Systems from Multilingual Natural Language Text

Lennart Wachowiak[1]
Centre for Translation Studies, University of Vienna, Austria

Christian Lang
Centre for Translation Studies, University of Vienna, Austria

Barbara Heinisch
Centre for Translation Studies, University of Vienna, Austria

Dagmar Gromann
Centre for Translation Studies, University of Vienna, Austria

Abstract

Terminological Concept Systems (TCS) provide a means of organizing, structuring and representing domain-specific multilingual information and are important to ensure terminological consistency in many tasks, such as translation and cross-border communication. While several approaches to (semi-)automatic term extraction exist, learning their interrelations is vastly underexplored. We propose an automated method to extract terms and relations across natural languages and specialized domains. To this end, we adapt pretrained multilingual neural language models, which we evaluate on term extraction standard datasets with best performing results and a combination of relation extraction standard datasets with competitive results. Code and dataset are publicly available.[2]

2012 ACM Subject Classification Computing methodologies → Information extraction; Computing methodologies → Neural networks; Computing methodologies → Language resources

Keywords and phrases Terminologies, Neural Language Models, Multilingual Information Extraction

Digital Object Identifier 10.4230/OASIcs.LDK.2021.22

Supplementary Material *Software (Source Code and Dataset)*: `https://github.com/Text2TCS/Towards-Learning-Terminological-Concept-Systems`
archived at `swh:1:dir:fff3183f35a3dd332e1d4a2cdc54d21259b4fae2`

Funding This research has been conducted within the Text2TCS project (`https://text2tcs.univie.ac.at/`) funded by European Language Grid (ELG) H2020 No. 825627.

1 Introduction

Terminological inconsistency represents one major source of misunderstanding in specialized communication. One vital measure to counteract such inconsistency is the creation of a TCS that represents concepts, their terms and interrelations. Thereby, it can be ensured that different parties in a communication, such as medical, political, and news teams in times of crisis, consistently refer to phenomena by utilizing the same words. Several approaches to automatically extract domain-specific terms, i.e., single- and multi-word sequences, from natural language text exist. Such methods rely on frequency-based to Wikipedia-link-based Automated Term Extraction (ATE) approaches [4]. ATE is further distinguished depending on whether it is performed on document or corpus level. However, to the best of our knowledge, no approaches to extract a full terminological concept system from multilingual texts have been proposed.

[1] Corresponding author
[2] `https://github.com/Text2TCS/Towards-Learning-Terminological-Concept-Systems`

3rd Conference on Language, Data and Knowledge (LDK 2021).
Editors: Dagmar Gromann, Gilles Sérasset, Thierry Declerck, John P. McCrae, Jorge Gracia, Julia Bosque-Gil, Fernando Bobillo, and Barbara Heinisch; Article No. 22; pp. 22:1–22:18
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

A TCS groups synonyms and equivalents across languages into a single concept and interrelates these concepts with a set of prespecified relations. A major distinction is made between hierarchical, i.e., generic and partitive, and non-hierarchical, e.g. activity and ownership, relations. While to the best of our knowledge there are no approaches for learning a TCS from text, neighboring fields can provide inspiration for the task at hand. For instance, entity linking represents the task of identifying and interlinking named entities based on information provided in text (e.g. [33]) and ontology learning also requires term and relation extraction (e.g. [31] build on deep learning).

In this work, we rely on recent developments of deep learning and especially the recent success of large pretrained multilingual language models. In our approach we split the task of learning a TCS from text into two sentence-level steps: 1) term extraction, and 2) relation extraction. We rely on adaptations of the pretrained multilingual language model XLM-RoBERTa (XLM-R) [9] for both steps and connect them in a pipeline. The first step reads the document sentence by sentence and assigns each word with one of three tags: term, term continuation, not a term. In the second step and with a different adaptation of XLM-R, we identify relations between terms building on a predefined set of hierarchical and non-hierarchical terminological relations.

Given the resulting information we automatically learn from text, this approach contributes to the topic of knowledge graphs as well as deep learning for Linguistic Linked Open Data (LLOD). Since XLM-R is highly multilingual, trained on 100 different languages, TCSs can be learned with the proposed approach from texts in any of those languages. Nevertheless, very few datasets for evaluating multilingual TCSs exist. For term extraction, we train and evaluate the system on the TermEval 2020 dataset [34] in English, French, and Dutch across four specialized domains as well as the English ACL RD-TEC 2.0 dataset [32]. For relation extraction, we rely on a combination of SemEval datasets [15, 20], a WCL hypernymy dataset [28], and manually annotated data we created, all of which are only available in English. To represent the resulting TCS, we currently rely on the ISO standard TermBase Exchange (TBX) format, however, the resulting information could be serialized in any adequate format. Methods for hosting terminological resources as LLOD have been proposed before (e.g. [11]).

In the next section, we present a brief theoretical introduction to a TCS and terminology, followed by an introduction to language models in Section 3. Section 4 details the data utilized to train and evaluate the proposed approach. Section 5 details the TCS learning method as well as the individual steps thereof, the results of which are presented in Section 6. We discuss the results in Section 7 and present related work in Section 8 prior to some concluding remarks.

2 Terminological Concept Systems

This section provides a brief overview of the field of terminology and TCS. It also states the relation typology utilized for our TCS learning approach.

2.1 Term, Concept and Terminology

Terminology can only be understood within the framework of specialized language or language for special purposes, which is defined as "**natural language** [...] used in communication between experts in a **domain** [...] and characterized by the use of specific linguistic means of expression" (emphasis in original) [1]. Thus, terminology refers to a set of concepts and their designations in a specialized field of knowledge (a language for special purposes). In

terminology studies, different schools of thought exist and we will follow the so-called General Theory of Terminology, where concepts and terms are differentiated, wherein concepts are considered abstractions of a set of physical or abstract entities and terms are their designation by linguistic means [3].

A designation can refer to a single- or multi-word term, a named entity, a symbol or even a formula. When talking about term extraction, in general extracting named entities and symbols is implicitly included. Concepts are rather vaguely referred to as units of thoughts and knowledge, however, we treat them as structuring means for synonymous terms. A concept system refers to the organization of concepts, and thereby also of knowledge.

Many ISO standards are based on this school of thought in terminology. The ISO standards address topics that range from the definition of terminology and terminology management to the representation of terminology in terminological databases. Among these standards is the TermBase eXchange (TBX) standard [2] that defines an industry representation format for exchanging terminological resources detailed below.

2.2 Relation Types

After identifying concepts, they can be analyzed and modeled by means of concept systems. Concept relations describe the link between different concepts. In the literature on terminology, different models for describing concept relations have been proposed, at times with a very large typology of relation types (e.g. [30]). On the highest level, relation types are generally classified into hierarchical and non-hierarchical relations.

Hierarchical relations connect a superordinate with a subordinate concept and are either generic or partitive. Generic relations exist "between two concepts when the intension of the subordinate concept includes the intension of the superordinate concept plus at least one additional delimiting characteristic" [3], such as "furniture" which serves as the superordinate concept for "desk". A lexical manifestation of this relation could, for instance, be "desk is a kind of furniture". Partitive relations exist when the superordinate concept relates to the entire object and the subordinate concept refers to its parts. For example, "root", "branch" and "stem" are parts of the superordinate concept "tree", or linguistically described a "stem is part of a tree".

Non-hierarchical relations are called associative in ISO 1087 [1], however, in the typology we adopt, an associative relation represents a thematic connection between two concepts that is not further specified. The number of non-hierarchical relations in the ISO 1087 standard [1] is rather small – sequential as superordinate to spatial, temporal and causal relations – and has been criticized for being inconsistent and ambiguous. Fortunately, a variety of relations have been proposed by different authors (e.g. [30]). In ontology learning and knowledge graph generation non-hierarchical relations also play a crucial role. However, the relation types generally vary significantly from one ontology or knowledge graph to another. While in the future we seek to map our typology to existing standard LLOD resources, for now we adopt relation types established in the terminology community and a consistent typology across domains and languages.

The relation types used in this study are derived from a literature review and were adapted to the needs of this research. The objective is to map semantic relations to this prespecified typology in order to ease the alignment between different TCSs resulting from our method across domains and languages, which consists of: *generic relation* (is a kind of, e.g. table is a kind of furniture) and *partitive relation* (is part of, e.g. roots are part of a tree), several non-hierarchical relations were included, that is, *spatial relation* (for objects

and their location, e.g. avalanche and mountain), *temporal relation* (for objects and their time or sequences, e.g. production and consumption), *causal relation* (for causes and their effect, e.g. accident and injury),

- Hierarchical relations:
  - *generic relation* the intension of the subordinate concept includes the intension of the superordinate concept plus one additional characteristic, e.g. "table is a kind of furniture"
  - *partitive relation* the subordinate concepts are parts of the superordinate concept, e.g. "roots are part of a tree"
- Non-hierarchical relations:
  - *activity relation* connects actors with an activity or an activity with its entity, e.g. "teacher activity schoolchildren" where the activity can be teaching,
  - *causal relation* connects causes and their effect, e.g. "accident causes injury",
  - *instrumental relation*: connects instruments and their use, e.g. "coffee machine instrumental coffee" since the former is utilized to make the latter,
  - *origination relation* connects an entity with its origin, e.g. "car origination factory" since the car originates from a factory,
  - *spatial relation* connects an entity with its location, e.g. "avalanche spatial mountain" since the former is located on the latter
  - *associative relation* provides a generic thematic connection between concepts, e.g. "lecturer associative education" since both are thematically associated to each other.

An associative relation can serve to model a connection between two loosely related concepts to which none of the other relation types applies. Apart from the symmetric associative relation, all relations are directed, e.g. for the instrumental relation, the direction is from instruments to their use and the partitive relation is directed from parts to whole. We initially treat synonymy as a symmetric relation in the relation extraction step, even though in the final output synonymy is not represented as relation, but as a set of synonyms to form a concept called terminological entry.

2.3 Representation in TBX

ISO 30042 [2] defines a framework to represent terminological data in a structured format called TBX, which aims at facilitating the exchange of terminological data for different purposes, including analysis, representation and dissemination. The users of TBX files include terminologists, translators or technical writers on the one hand, and computer applications, such as computer-assisted translation tools and authoring software, on the other. It is a flexible format that allows for user-defined relation types. As the de-facto standard for terminological resource representation in industry and academia, we opted for TBX as our initial output format, but intend to accommodate RDF directly and not only by way of conversion of TBX to RDF [11] in the near future.

3 Language Models

Most of the recent progress in natural language processing can be traced back to transformer-based pretrained language models. This type of transfer learning utilizes deep neural networks based on the transformer architecture [38]. In a first stage called pretraining, the network learns to predict a masked word given its context, a task for which large amounts of training data are available. In the second stage called fine-tuning, the pretrained network is used

again and is trained for a specific task like classification by adding additional layers on top of the model, while making use of the previously learned rich language representations. A frequently used English language model is BERT [10], for which also multilingual variants exist that have been pretrained on corpora in multiple languages, e.g. multilingual BERT or XLM-R [9]. XLM-R uses the enhanced training paradigm introduced by RoBERTa [26] while being pretrained on a CommonCrawl dataset in 100 different languages. Due to the pretraining, multilingual models can be fine-tuned in one language and show strong zero-shot performances in another language, for which no training samples were provided during fine-tuning.

4 Data

The data used for training and evaluation was compiled from multiple datasets. In order to effectively use the data to train our model for the specified tasks, some pre-processing as well as manual creation of additional data was necessary.

4.1 Term Extraction Data

For term extraction, we used the Annotated Corpora for Term Extraction Research (ACTER)[3] that was also used in the TermEval 2020 challenge [34]. This provided us with hand-annotated data and a good baseline to evaluate our systems. The data comes in the form of raw text documents divided into four domains (corruption, dressage/equitation, wind energy, heart failure) and a single document per domain containing all terms that have been identified in the text documents by language experts. The terms are provided with the same surface-form as they appear in the texts, so each term-list may contain several morphological variations of a term. All terms are provided in lower-case. Additionally, the data is provided in three languages, namely English, French, and Dutch. At the time of writing, the most recent version of the dataset was version 1.4, which did not provide inline annotations. Since our training is performed on sentences, we needed to annotate each sentence in the provided text documents with the terms from the corresponding term document.

To annotate the texts, we first split the documents into sentences using the spaCy sentencizer [21] and tokenized each sentence using sacremoses[4], a tokenizer written in Python. Subsequently, each individual sentence was annotated with terms from the term document of the corresponding domain. Only terms that had a full match with any word or word sequence composing each sentence were annotated. This way it was possible to create an inline annotation from the raw data. In order to allow a comparison with the TermEval results we followed the train-val-test-split of the TermEval 2020 challenge and used the corruption and wind energy domains as training, the dressage (equitation) domain as validation, and the heart failure domain as test data set. With this train-val-test-split, around 10,000 sentences per language were used for supervised training, while the test set contained approximately 2,000 sentences. The exact word and term count per language is represented in Table 2.

We also trained separate models using the ACL RD-TEC 2.0 dataset [32] to verify if our approach would work on other datasets. The ACL RD-TEC 2.0 dataset provides high-quality inline annotations of 471 scientific abstracts by two human annotators. In total more than 2,200 sentences were annotated with 6,818 terms. Annotator 1 annotated 900 sentences and

[3] https://github.com/AylaRT/ACTER
[4] https://github.com/alvations/sacremoses

Annotator 2 1,301 sentences. Since no split is proposed by the dataset authors, we split the data into 60% training data, 20% validation data, and 20% test data for each of the two annotators respectively. This resulted in 540 training sentences and 180 val/test sentences for Annotator 1 and 780 training sentences and 260/261 val/test sentences for Annotator 2 respectively. Since the dataset is inline annotated, the terms from each sentence could be easily extracted with an XML Parser. Additionally, unlike TermEval, capitalization of terms is maintained both for the training and validation/test data.

4.2 Relation Extraction Data

For relation extraction, we combined training data from two SemEval tasks to obtain more training examples with a higher diversity of relation types: SemEval 2007 Task 4 [15] and SemEval 2010 Task 8 [20]. We then mapped the relation types of these datasets to the relation types defined for our TCS (see Section 2.2). Since these two datasets lack generic relations, we additionally utilized the manually created WCL 1.2 dataset [28] and automatically annotated the terms in the ACTER dataset texts with generic relations and synonyms, which were also not represented in the other datasets, by relying on WordNet relations. Furthermore, we extracted acronyms and their long unabbreviated forms as synonyms from the ACTER texts by adapting the regex-based acronym extraction method proposed by Azimi et al. [6]. We further added to the data by manually annotating around 200 acronym-term pairs from ACTER with relations other than synonymy and another 271 sample sentences from the Common Crawl News Corpus[5] with term pairs that show no relation at all, i.e., negative samples to be classified as "none". All samples of the resulting relation dataset are in English. Statistics regarding the relation type distribution can be seen in Figure 1.

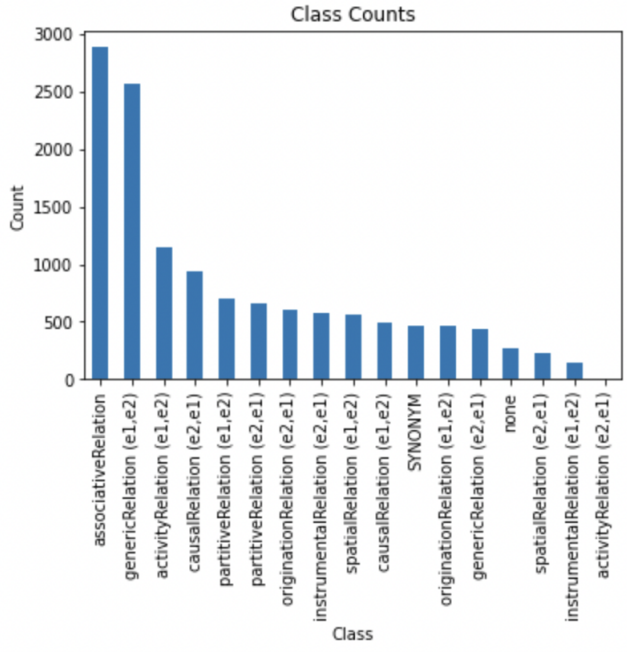


Figure 1 Number of samples for each relation type.

[5] https://commoncrawl.org/2016/10/news-dataset-available/

For the data originating from SemEval we use the original train-test split, while from the additional data we take 20% for testing. An additional 20% from all resulting training data is used for validation.

5 Method

We will first describe the overall architecture of the proposed TCS learning pipeline. Subsequently, we introduce our term extraction and relation extraction models.

5.1 Architecture

The TCS extraction pipeline, as shown in Figure 2, takes text documents as input and outputs terminologies in the TBX format as well as in the form of a graph visualization, an example of which is shown in Figure 3. Due to input length restrictions of current language models, terms as well as their relations are extracted on sentence-level basis. Thus, as the first and only preprocessing step, the input document is split into sentences using the rule-based sentence segmentation component provided by the software library spaCy [21].

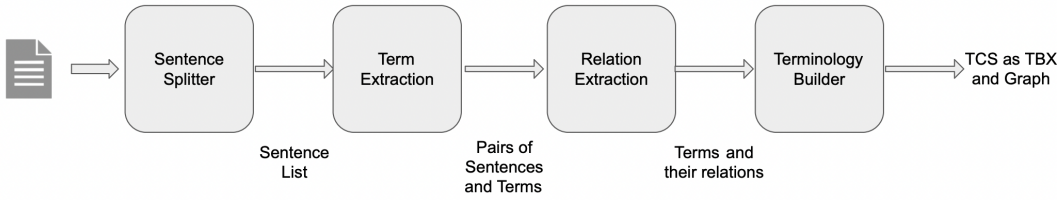


Figure 2 TCS learning pipeline.

In a second step, terms are extracted from each sentence using the neural network described in Section 5.2. After the term extraction step, we end up with pairs of sentences and the terms they contain according to the model.

Based on this data, all possible pairs of terms are computed per sentence. These pairs together with their respective context, i.e., the sentence which contains them, are then fed one after the other to the second neural network, described in Section 5.3, which outputs whether or not there is a relation between the terms and, if so, which relation type exactly.

As the fourth step, these term pairs and their corresponding relations are used to create a terminological concept system. Therefore, terms with synonym relations are merged into concepts with a unique identifier. The extracted relations, which at this point are still between specific terms, are mapped to the newly created concepts. Through this process it is possible that self-referential relations as well as duplicate relations are created, which are subsequently deleted. Moreover, it is possible that there are multiple different relations between the same two concepts, however, only one is represented in the final TCS. To provide more insight into this process to the final user, we show the extraction network's classification confidence in the output. Lastly, the resulting TCS is represented in the TBX format and as a graph utilizing the Graphviz library [13].

5.2 Term Extraction Model

For term extraction, we take advantage of the multilingual pretrained language model XLM-R in its base size made available by the transformers library [40]. The input provided to the model consists of full sentences and is based on the data described in Section 4.1. The output

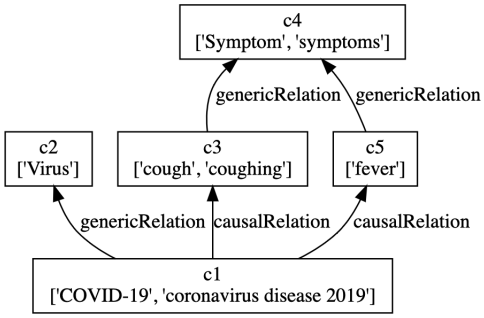


Figure 3 Example of a possible visualization with Graphviz.

labels are given on a word basis, i.e., each word is either tagged as "B-T" (beginning of a term), "T" (continuation of a term), or "n" (no term). For instance, the labels for the sentence "We meta-analyzed mortality using random-effect models" are "n", "B-T", "B-T", "n", "B-T", "T". For classification into these three classes we use a single fully connected layer which uses the representations created by XLM-R for the individual words as input. Since XLM-R tokenizes the input on a subword level, we obtain labels for these subword units which have to be mapped back to the original input. Training was conducted using the Adam optimizer with a learning rate of 2e-5, a batch size of 8, and a validation every 100 steps allowing us to load the best performing model at the end of the training procedure, which consisted of 10 epochs overall.

5.3 Relation Extraction Model

As with term extraction, we fine-tune the pretrained XLM-R for the relation extraction task. The data used is described in Section 4.2. The input of the model consists of an entity pair followed by a contextualizing sentence containing both entities, for instance, "cough. Covid-19. The cough was caused by Covid-19." The model classifies the representation of the whole sequence created by XLM-R as input with a fully connected layer into one of the relations presented in Section 2.2. For directional relations two classes are available, so that the given example input would, for instance, be classified as `causalRelation(e2,e1)`. The model was trained for 9 epochs utilizing the Adam optimizer with a learning rate of 2e-5 and a batch size of 32.

6 Results

Since no datasets for a full TCS evaluation are available as of yet, we evaluate our model on the described datasets separately and present the results below.

6.1 Term Extraction

For the term extraction step, we evaluated our model in comparison to the best preforming models of the TermEval 2020 shared task in terms of precision, recall and F1 score as shown in Table 1. These metrics are calculated on the basis of the available annotation in the original ACTER dataset, where we opted for the more comprehensive list of terms including named entities. Since different combinations of training and test languages might have an impact on the overall performance, we report on these combinations in Table 1. As in TermEval 2020 we use the heart failure domain as hold-out test set. The baseline for English and

French is provided by [19], who used monolingual neural language models to predict whether a given phrase is a term provided some context. Thus, other combinations as tested with our multilingual model are not available with monolingual models. The baseline for Dutch is provided by a bidirectional LSTM with GLOVE[6]. The overall best results are marked in bold for each test language. For the ACL RD-TEC 2.0 corpus no baseline is available and the data split into 60% training, 20% validation, and 20% test data was chosen by us as no split was suggested by the authors of the dataset. The results are also available in Table 1.

Table 1 Test set results of our term extraction model on two different datasets evaluated on different langauge combinations.

Dataset	**Training**	**Test**	**Token Classifier**			**Previous SOTA**		
			Prec	Rec	F1	Prec	Rec	F1
TermEval 2020	EN	EN	54.9	62.2	**58.3**	34.8	70.9	46.7
TermEval 2020	FR	EN	56.7	36.2	44.2			
TermEval 2020	NL	EN	55.3	61.8	**58.3**			
TermEval 2020	ALL	EN	54.4	58.2	56.2			
TermEval 2020	EN	FR	65.4	51.4	**57.6**			
TermEval 2020	FR	FR	68.7	43.0	52.9	44.2	51.5	48.1
TermEval 2020	NL	FR	62.3	48.5	54.5			
TermEval 2020	ALL	FR	49.4	55.3	55.0			
TermEval 2020	EN	NL	67.9	71.7	**69.8**			
TermEval 2020	FR	NL	69.2	55.2	61.4			
TermEval 2020	NL	NL	71.4	67.8	69.6	18.9	18.6	18.7
TermEval 2020	ALL	NL	70.0	65.8	67.8			
ACL (An.1)	EN	EN	74.4	77.2	75.8			
ACL (An.2)	EN	EN	80.1	79.3	80.0			

It can be seen from Table 1 that our solution outperforms the TermEval 2020 baseline in all three languages. However, it is interesting to see that mixed training and test languages achieve best results. As a matter of fact, the model trained on English achieved best results not only when tested on English, but also when tested on French and Dutch. When the test language was English, training on Dutch achieved equivalent results to training on English. A general assumption would be that training on the language that is being tested or training on all available languages should perform best, an assumption that could not be confirmed in this experiment. Furthermore, a substantial difference in recall behavior can be observed from one and the same model, which can also be observed with the monolingually pretrained baseline models even though they achieve a competitive or even higher recall. This suggests differences in term type across the three languages. This observation can also be confirmed in the validation set performance reported in Table 2, where French as training and validation set performs significantly worse. Validation results also show some performance differences to the test performances.

The models trained on the ACL RD-TEC 2.0 corpus show an even stronger performance with an F1 score of 75.8 and 80.0 for Annotator 1 and 2 respectively. Moreover, the scores for precision and recall of the two resulting models are nearly perfectly balanced. The validation scores, reported in Table 2, are consistent with the test scores.

[6] No system description paper was submitted for this approach after participation in the challenge.

Table 2 Train, validation and test split by word count (W_{lang}) and term count (T_{lang}) per language (left) and validation performance of token classifier (right).

	Train	Val	Test
W_{en}	97,145	51,470	45,788
T_{en}	2,708	1,575	2,585
W_{fr}	106,792	53,316	46,751
T_{fr}	2,185	1,183	2,423
W_{nl}	96,887	50,882	47,888
T_{nl}	2,540	1,546	2,257

Training	EN Val	FR Val	NL Val
EN (TermEval)	55.6	45.3	60.5
FR (TermEval)	41.9	33.6	49.6
NL (TermEval)	54.6	47.7	57.8
ALL (TermEval)	50.0	40.4	51.5
EN (ACL An.1)	75.5	/	/
EN (ACL An.2)	79.3	/	/

In terms of term type, the models trained on TermEval 2020 are able to handle acronyms well, which might be due to the fact that much of the training data was based on rather technical documents like scientific abstracts. However, if acronyms are part of the term, e.g. “LV strain rate”, there was a high number of false negatives. Equally apostrophes in named entities represented a challenge, e.g. “Chaga’s disease”. We could observe a tendency of the model to split particularly long multi-word sequences (more than five words), e.g. “resynchronization reverses remodeling in systolic left ventricular dysfunction”. We made similar observations when manually evaluating the model trained and tested on Annotator 1 of ACL RD-TEC 2.0 dataset. While performance on acronym extraction was generally good, if acronyms were part of a term, it likely resulted in a false negative. It can be observed, that false positives often correlate with the false negatives, as the model extracts only parts of the original term or splits the longer terms, e.g. “LRE project SmTA double check” is extracted as “LRE” and “SmTA double check”. It was especially noticeable that the model had difficulties extracting terms containing their acronym or the expansion of an acronym in parentheses, e.g. “machine translation (MT) systems”. This issue also extended to other terms containing parentheses, such as “document descriptors (keywords)”. In fact, not a single example of terms containing parentheses was extracted. Similar to the model trained on TermEval 2020 data, the model trained on the ACL RD-TEC 2.0 data showed a general tendency for extracting shorter terms, with the largest group of false negatives being terms composed of four or more words (41 out of 124 examples).

The model trained on the TermEval 2020 dataset turned out to be highly efficient in terms of training time. Looking at the epochs required to reach the best score on the validation set, we can observe that in most cases the token classifier model requires not even a single training epoch. Training with the English dataset required 300 steps with a full epoch consisting of 432 steps. The model trained on French was the only model with its best performance being reached during the second epoch after 700 steps, while a full epoch consists of 437 steps. The model trained on Dutch performed best after 400 steps while one epoch takes 553 steps. The multilingual model converged the quickest needing only 200 steps whereas a full epoch consists of 1,421 steps. The models trained on the ACL RD-TEC 2.0 dataset need more epochs and achieve their highest scores after 3 and 5 epochs respectively. However, due to the lower training set size of the ACL RD-TEC 2.0 corpus this also corresponds to less than 500 steps, thus, being similar with the training times reported for the model trained with TermEval 2020 data.

6.2 Relation Extraction

The trained model achieves a weighted averaged F1 score of 87% with a precision of 87% and a recall of 87% on the hold-out test set. The confusion matrix in Figure 4 and Table 3 show which classes were learned best. Only the activity relation from entity 2 to entity 1 was not

Table 3 Test performance of the relation classifier and number of test samples.

Relation Type	Precision	Recall	F1	Test samples
synonymy	0.85	0.76	0.80	89
activityRelation (e1,e2)	0.93	0.97	0.95	293
activityRelation (e2,e1)	0.00	0.00	0.00	2
associativeRelation	0.90	0.92	0.91	783
causalRelation (e1,e2)	0.90	0.95	0.92	135
causalRelation (e2,e1)	0.92	0.91	0.91	222
genericRelation (e1,e2)	0.90	0.93	0.92	533
genericRelation (e2,e1)	0.46	0.41	0.43	91
instrumentalRelation (e1,e2)	0.72	0.68	0.70	34
instrumentalRelation (e2,e1)	0.85	0.88	0.86	144
none	0.69	0.44	0.54	70
originationRelation (e1,e2)	0.83	0.89	0.86	116
originationRelation (e2,e1)	0.84	0.83	0.83	165
partitiveRelation (e1,e2)	0.90	0.85	0.87	176
partitiveRelation (e2,e1)	0.77	0.77	0.77	168
spatialRelation (e1,e2)	0.90	0.91	0.91	169
spatialRelation (e2,e1)	0.90	0.82	0.86	44

learned at all given the current training data as the class consists of overall less than 10 data points. Activity relations are usually directed from actor to activity, which was also the case in our dataset, i.e., the actor was mostly mentioned first (e1) and the activity second (e2), with less than 10 exceptions where the actor was mentioned second. The only other relations with an F1 score lower than 0.7 are the none-relation and the generic relation from e2 to e1, which also can be traced back to relatively small amounts of training data as well as a high confusion with the same relation in the opposite direction in case of the generic relation. For the synonymy relation, the classification of synonym pairs including an acronym works well, while the relation of two longer sequence (not shortened) pairs is often confused as a generic relation. Many of the other relations, especially those supported by large amounts of training data, achieve high F1 scores of up to 95%. Furthermore, we can observe very balanced precision and recall scores for all relations.

7 Discussion

Currently, the proposed pipeline fully operates on a sentence-level. In the future we plan to extend the architecture so that the model can extract relations which span over the whole document. This could be achieved by models trained on an appropriate dataset containing such relations. However, currently such datasets are rare and available ones are either domain-specific, very small, and/or focus on named entities [25, 42]. Another option to extract document-wide relations is to add a model to the pipeline which makes predictions about the relation between two words independent from any context in which they appear, something for which, for instance, approaches for hierarchical relations exist [39]. Such a model could be applied to all possible term-pairs, however, due to the missing contexts only limited effectiveness can be expected.

A problem of pipeline approaches is that errors from earlier pipeline steps are propagated to the later components. In the case of the TCS extraction pipeline, wrongly extracted terms are sent to the relation extraction component which tries to establish a relation to

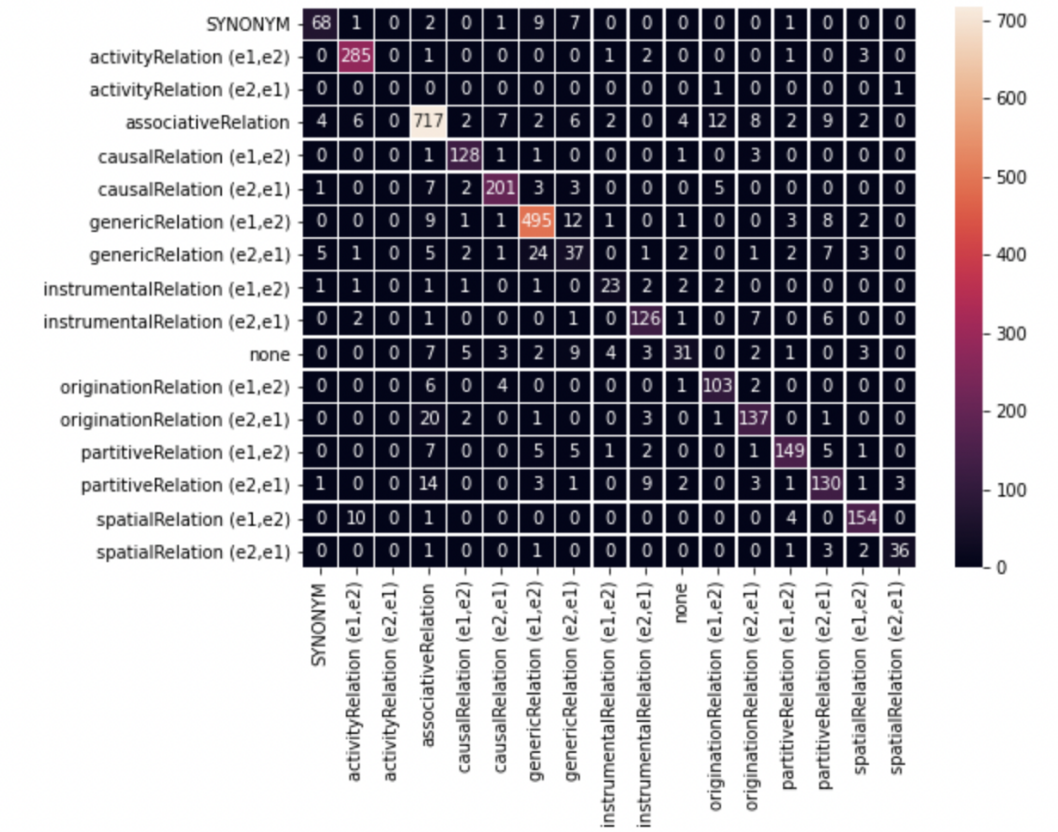


Figure 4 Confusion Matrix for the relation extraction model on the test set.

other terms. This problem can potentially be solved by joint models, which learn to extract terms and their relations together, as was already done in the case of named entities and very limited domain-specific entities and their relations (e.g. [22, 29]). Such models, however, require datasets that annotate terms and their corresponding relations in the same texts, which is something that is currently not available, but something that we are aiming to make available in the future.

8 Related Work

Since the proposed pipeline to automatically learn TCS from text relies on two intermediate steps, term and relation extraction, as well as their combination, we separate the related work into the individual steps as well as approaches joining both steps.

8.1 Term Extraction

An initial classification of ATE methods into statistical, linguistic or hybrid has recently been refined to methods based on term occurrence frequencies, occurrence contexts, domain-specific corpora combined with general language corpora, topic modeling, and those utilizing Wikipedia (see [4] for an overview). Methods are additionally categorized by the type of context, i.e., corpus-level (e.g. [4, 45]) and document-level (e.g. [37]) settings. However, neural ATE methods frequently operating on sentence-level cannot be easily accommodated by these classifications.

The approach most closely related to ours, which also provided our baseline [19], utilized RoBERTa [26] for English and CamemBERT [27] for French and won the TermEval 2020 challenge. In their work, pretrained language models clearly outperformed a classification method based on a variety of features, such as statistical descriptors. They, however, train their model using pairs of context sentences and possible candidate terms, which are based on all possible n-grams contained in a given context sentence, a procedure much slower than our proposed token-level classifier. A recently published approach [36] relies on LSTM, GRU and BERT embeddings and achieves high F1 scores for ATE of Lithuanian terms in the cybersecurity domain. Several approaches build on word embeddings to perform ATE on specific domains, such as medicine (e.g. [7]), or to separate general-language from domain-specific embeddings [18]. In contrast, our model performs ATE on four domains and in three languages utilizing a pretrained language.

8.2 Relation Extraction

Relation extraction describes the supervised task of classifying a relation given two entities and a context. Most work and datasets in the field focus on sentence-level relation extraction [15, 20, 44] with only some exceptions providing relations over longer text spans [42]. Current state-of-the-art approaches for such datasets usually rely on either transformer-based architectures [41, 47] or graph-based neural networks [17, 43].

8.3 Joint Term and Relation Extraction

While to the best of our knowledge no approaches exist to automatically extract terminological concept systems from text, there is an entire research field on connecting terminological information with ontologies, thereby providing relational information to terms. Methods for modeling terminological information as ontologies are generally called terminological ontologies (e.g. [24]). Approaches that model terminological information in relation to ontologies are generally called ontology-terminology models (e.g. [35, 16]). One approach in this direction that is probably most closely related to the one proposed in this paper is TERMINAE [5], a platform that utilizes traditional NLP tools and methods to propose term candidates and relations to users for manual editing by building on terminology engineering principles and findings from ontology learning. While a very interesting method and platform, the approach neither commits to a specific typology of relations nor seeks to provide a fully automated solution. Thus, to broaden the scope of this discussion on related work and consider related fields that directly inspire this joint task, we will discuss two additional research directions. First, we present approaches on entity and relation extraction. Second, we relate to selected ontology learning approaches.

Joint entity and relation extraction (e.g. [46]) is the task of identifying named entities in text and detecting their semantic relations. Approaches to this task range from joining a bidirectional LSTM for term extraction with a CNN for relation extraction [46] to utilizing a Graph Convolutional Network [12]. This idea of joining recurrence and convolution operations is taken up again by Geng et al. [14]. The approach probably most similar to ours is that of Quiao et al. [33] who utilize BERT for joint entity and relation extraction in the agricultural domain. However, our approach has been applied across several domains and languages. In addition, named entities are a subcategory of single- and multi-word terms, where the latter is considerably more challenging. The type of relation is also frequently restricted to lexical-semantic relations, such as synonymy or hypernymy, specific semantic relations, such as the temporal relation, or information in a specific domain, e.g. agriculture.

Ontology learning (e.g. [31, 8]) is the task of automatically extracting knowledge from text, starting with terms which are organized to form concepts, their interrelations which are organized hierarchically and non-hierarchically, and finally axioms. Petrucci et al. [31] utilize Neural Machine Translation (NMT) on a synthetically generated dataset to learn Description Logic formulas from natural language sentences. In contrast, our approach operates on non-synthetic, real-life datasets. Few other approaches utilize deep learning for ontology learning (see [23] for an overview).

9 Conclusion

As a first step to approach fully automated TCS learning from multilingual text, we propose adaptations of pretrained language models to perform term and relation extraction in a pipeline approach. While a multilingual, cross-domain dataset for term extraction exists, we had to accumulate and extend several relation extraction datasets to accommodate a common terminological relation typology. Term extraction results substantially outperform previous results and the relation extraction model achieves competitive results, even though no baseline comparison was available for exactly these relation types.

As a next step we will manually create a full evaluation dataset for TCS across domains and languages to provide a better evaluation scenario for the proposed approach. Additionally, the model currently exclusively extracts information from sentences, whereby several global relations beyond the sentential level will be lost, especially synonymy and generic relations. We thus currently evaluate methods for achieving document-level TCS learning. Lastly, we will extend the set of covered relations by including data for temporal, property, and ownership relations.

References

1 ISO 1087:2019. Terminology work and terminology science – Vocabulary. Standard, International Organization for Standardization, Geneva, CH, 2019.

2 ISO 30042:2019. Management of terminology resources – TermBase eXchange (TBX). Standard, International Organization for Standardization, Geneva, CH, 2019.

3 ISO 704:2009. Terminology work – Principles and methods. Standard, International Organization for Standardization, Geneva, CH, 2009.

4 Nikita Astrakhantsev. ATR4S: toolkit with state-of-the-art automatic terms recognition methods in scala. *Language Resources and Evaluation*, 52(3):853–872, 2018.

5 Nathalie Aussenac-Gilles, Sylvie Despres, and Sylvie Szulman. The terminae method and platform for ontology engineering from texts, 2008.

6 Sasan Azimi, Hadi Veisi, and Reyhaneh Amouie. A method for automatic detection of acronyms in texts and building a dataset for acronym disambiguation. In *2019 5th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)*, pages 1–4. IEEE, 2019. doi:10.1109/ICSPIS48872.2019.9066084.

7 Matthias Bay, Daniel Bruneß, Miriam Herold, Christian Schulze, Michael Guckert, and Mirjam Minor. Term extraction from medical documents using word embeddings. In *4th IEEE Conference on Machine Learning and Natural Language Processing (MNLP 2020)*. IEEE Computer Society, 2020. URL: http://www.wi.cs.uni-frankfurt.de/webdav/publications/TLDIA_Paper_IEEE_CRC.pdf.

8 Philipp Cimiano and Johanna Völker. text2onto. In *International conference on application of natural language to information systems*, pages 227–238. Springer, 2005.

9 Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, July 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.747.

10 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi:10.18653/v1/N19-1423.

11 Maria Pia di Buono, Philipp Cimiano, Mohammad Fazleh Elahi, and Frank Grimm. Terme-à-LLOD: Simplifying the conversion and hosting of terminological resources as linked data. In Maxim Ionov, John P. McCrae, Christian Chiarcos, Thierry Declerck, Julia Bosque-Gil, and Jorge Gracia, editors, *Proceedings of the 7th Workshop on Linked Data in Linguistics (LDL-2020)*, pages 28–35, Marseille, France, May 2020. European Language Resources Association. URL: https://www.aclweb.org/anthology/2020.ldl-1.5.

12 Tsu-Jui Fu, Peng-Hsuan Li, and Wei-Yun Ma. GraphRel: Modeling text as relational graphs for joint entity and relation extraction. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1409–1418, Florence, Italy, July 2019. Association for Computational Linguistics. doi:10.18653/v1/P19-1136.

13 Emden R. Gansner and Stephen C. North. An open graph visualization system and its applications to software engineering. *Software – Practice and Experience*, 30(11):1203–1233, 2000.

14 Zhiqiang Geng, Yanhui Zhang, and Yongming Han. Joint entity and relation extraction model based on rich semantics. *Neurocomputing*, 429:132–140, 2021. doi:10.1016/j.neucom.2020.12.037.

15 Roxana Girju, Preslav Nakov, Vivi Nastase, Stan Szpakowicz, Peter Turney, and Deniz Yuret. SemEval-2007 task 04: Classification of semantic relations between nominals. In Eneko Agirre, Lluís Màrquez, and Richard Wicentowski, editors, *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 13–18, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/S07-1003.

16 Dagmar Gromann. A model and method to terminologize existing domain ontologies. In *Terminology and Knowledge Engineering 2014*, pages 10–p, 2014.

17 Zhijiang Guo, Yan Zhang, and Wei Lu. Attention guided graph convolutional networks for relation extraction. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 241–251, Florence, Italy, July 2019. Association for Computational Linguistics. doi:10.18653/v1/P19-1024.

18 Anna Hätty, Dominik Schlechtweg, Michael Dorna, and Sabine Schulte im Walde. Predicting degrees of technicality in automatic terminology extraction. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2883–2889, Online, July 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.258.

19 Amir Hazem, Mérieme Bouhandi, Florian Boudin, and Beatrice Daille. TermEval 2020: TALN-LS2N system for automatic term extraction. In Béatrice Daille, Kyo Kageura, and Ayla Rigouts Terryn, editors, *Proceedings of the 6th International Workshop on Computational Terminology*, pages 95–100, Marseille, France, 2020. European Language Resources Association. URL: https://www.aclweb.org/anthology/2020.computerm-1.13.

20 Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In Katrin Erk and Carlo Strapparava, editors, *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/S10-1006.

21 Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spaCy: Industrial-strength Natural Language Processing in Python, 2020. doi:10.5281/zenodo.1212303.

22 Bin Ji, Jie Yu, Shasha Li, Jun Ma, Qingbo Wu, Yusong Tan, and Huijun Liu. Span-based joint entity and relation extraction with attention-based span-specific and contextual semantic representations. In Donia Scott, Nuria Bel, and Chengqing Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics*, pages 88–99, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. doi:10.18653/v1/2020.coling-main.8.

23 Ahlem Chérifa Khadir, Hassina Aliane, and Ahmed Guessoum. Ontology learning: Grand tour and challenges. *Computer Science Review*, 39, 2021. doi:10.1016/j.cosrev.2020.100339.

24 Javier Lacasta, Javier Nogueras-Iso, and Francisco Javier Zarazaga Soria. *Terminological Ontologies: Design, Management and Practical Applications*, volume 9. Springer Science & Business Media, 2010.

25 Jiao Li, Yueping Sun, Robin Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn Mattingly, Thomas Wiegers, and Zhiyong lu. Biocreative v cdr task corpus: a resource for chemical disease relation extraction. *Database*, page baw068, 2016. doi:10.1093/database/baw068.

26 Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019. arXiv:1907.11692.

27 Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. CamemBERT: a tasty French language model. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online, 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.645.

28 Roberto Navigli, Paola Velardi, and Juana Maria Ruiz-Martínez. An annotated dataset for extracting definitions and hypernyms from the web. In Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May 2010. European Language Resources Association (ELRA). URL: http://www.lrec-conf.org/proceedings/lrec2010/pdf/20_Paper.pdf.

29 Tapas Nayak and Hwee Tou Ng. Effective modeling of encoder-decoder architecture for joint entity and relation extraction. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8528–8535. AAAI Press, 2020. URL: https://aaai.org/ojs/index.php/AAAI/article/view/6374.

30 Anita Nuopponen. Tangled web of concept relations. concept relations for iso 1087-1 and iso 704. In *Terminology and Knowledge Engineering 2014*, Berlin, Germany, 2014. Association for Computational Linguistics. URL: https://hal.archives-ouvertes.fr/hal-01005882.

31 Giulio Petrucci, Marco Rospocher, and Chiara Ghidini. Expressive ontology learning as neural machine translation. *Journal of Web Semantics*, 52:66–82, 2018. doi:10.1016/j.websem.2018.10.002.

32 Behrang QasemiZadeh and Anne-Kathrin Schumann. The ACL RD-TEC 2.0: A language resource for evaluating term extraction and entity recognition methods. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1862–1868, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL: https://www.aclweb.org/anthology/L16-1294.

33 Bo Qiao, Zhuoyang Zou, Yu Huang, Kui Fang, Xinghui Zhu, and Yiming Chen. A joint model for entity and relation extraction based on bert. *Neural Computing and Applications*, pages 1–11, 2021. doi:10.1007/s00521-021-05815-z.

34 Ayla Rigouts Terryn, Veronique Hoste, Patrick Drouin, and Els Lefever. TermEval 2020: Shared task on automatic term extraction using the annotated corpora for term extraction research (ACTER) dataset. In Béatrice Daille, Kyo Kageura, and Ayla Rigouts Terryn, editors, *Proceedings of the 6th International Workshop on Computational Terminology*, pages 85–94, Marseille, France, May 2020. European Language Resources Association. URL: https://www.aclweb.org/anthology/2020.computerm-1.12.

35 Christophe Roche, Marie Calberg-Challot, Luc Damas, and Philippe Rouard. Ontoterminology: A new paradigm for terminology. In *International Conference on Knowledge Engineering and Ontology Development*, pages 321–326, 2009.

36 Aivaras Rokas, Sigita Rackevičienė, and Andrius Utka. Automatic extraction of lithuanian cybersecurity terms using deep learning approaches. In *Human Language Technologies–The Baltic Perspective*, volume 328, pages 39–46. IOS Press, 2020. doi:10.3233/FAIA200600.

37 Antonio Šajatović, Maja Buljan, Jan Šnajder, and Bojana Dalbelo Bašić. Evaluating automatic term extraction methods on individual documents. In Agata Savary, Carla Parra Escartín, Francis Bond, Jelena Mitrović, and Verginica Barbu Mititelu, editors, *Proceedings of the Joint Workshop on Multiword Expressions and WordNet (MWE-WN 2019)*, pages 149–154, Florence, Italy, 2019. Association for Computational Linguistics. doi:10.18653/v1/W19-5118.

38 Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, undefinedukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc.

39 Lennart Wachowiak, Christian Lang, Barbara Heinisch, and Dagmar Gromann. CogALex-VI shared task: Transrelation - a robust multilingual language model for multilingual relation identification. In Rong Xiang, Emmanuele Chersoni, Luca Iacoponi, and Enrico Santus, editors, *Proceedings of the Workshop on the Cognitive Aspects of the Lexicon*, pages 59–64, Online, 2020. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/2020.cogalex-1.7.

40 Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771, 2019. arXiv:1910.03771.

41 Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. LUKE: Deep contextualized entity representations with entity-aware self-attention. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454, Online, November 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-main.523.

42 Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. DocRED: A large-scale document-level relation extraction dataset. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 764–777, Florence, Italy, July 2019. Association for Computational Linguistics. doi:10.18653/v1/P19-1074.

43 Shuang Zeng, Runxin Xu, Baobao Chang, and Lei Li. Double graph based reasoning for document-level relation extraction. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1630–1640, Online, November 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-main.127.

44 Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. Position-aware attention and supervised data improve slot filling. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi:10.18653/v1/D17-1004.

45 Ziqi Zhang, Jose Iria, Christopher Brewster, and Fabio Ciravegna. A comparative evaluation of term recognition algorithms. In Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, and Daniel Tapias, editors, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, May 2008. European Language Resources Association (ELRA). URL: http://www.lrec-conf.org/proceedings/lrec2008/pdf/538_paper.pdf.

46 Suncong Zheng, Yuexing Hao, Dongyuan Lu, Hongyun Bao, Jiaming Xu, Hongwei Hao, and Bo Xu. Joint entity and relation extraction based on a hybrid neural network. *Neurocomputing*, 257:59–66, 2017. doi:10.1016/j.neucom.2016.12.075.

47 Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. Document-level relation extraction with adaptive thresholding and localized context pooling. *CoRR*, abs/2010.11304, 2020. arXiv:2010.11304.

Encoder-Attention-Based Automatic Term Recognition (EA-ATR)

Sampritha H. Manjunath
Insight Centre for Data Analytics, National University of Ireland, Galway, Ireland

John P. McCrae
Insight Centre for Data Analytics, National University of Ireland, Galway, Ireland

Abstract

Automated Term Recognition (ATR) is the task of finding terminology from raw text. It involves designing and developing techniques for the mining of possible terms from the text and filtering these identified terms based on their scores calculated using scoring methodologies like frequency of occurrence and then ranking the terms. Current approaches often rely on statistics and regular expressions over part-of-speech tags to identify terms, but this is error-prone. We propose a deep learning technique to improve the process of identifying a possible sequence of terms. We improve the term recognition by using Bidirectional Encoder Representations from Transformers (BERT) based embeddings to identify which sequence of words is a term. This model is trained on Wikipedia titles. We assume all Wikipedia titles to be the positive set, and random n-grams generated from the raw text as a weak negative set. The positive and negative set will be trained using the Embed, Encode, Attend and Predict (EEAP) formulation using BERT as embeddings. The model will then be evaluated against different domain-specific corpora like GENIA – annotated biological terms and Krapivin – scientific papers from the computer science domain.

2012 ACM Subject Classification Information systems → Top-k retrieval in databases; Computing methodologies → Information extraction; Computing methodologies → Neural networks

Keywords and phrases Automatic Term Recognition, Term Extraction, BERT, EEAP, Deep Learning for ATR

Digital Object Identifier 10.4230/OASIcs.LDK.2021.23

Funding This publication has emanated from research supported in part by a research grant from Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289_P2 (Insight_2).

Acknowledgements We would like to thank the reviewers for helpful comments and insightful feedback.

1 Introduction

Terms are an important aspect in many applications that deal with natural languages such as search engines, automatic thesaurus construction [3], information extraction [9], automatic abstraction [19], machine translation and ontology [17] and glossary population.

There are many methods to achieve the ATR task which include rule-based methods and machine learning methods (data-driven) [18]. Rule-based methods need a set of pre-defined rules for each task which needs deep knowledge of the domain and is often difficult to maintain. Machine learning-based methods, on the other hand, have a significant effect on existing classification activities, and experiments have shown considerable improvement. The classical approach includes two steps, first feature extraction using methods like bag-of-words and second, then using classification algorithms like support vector machines (SVM) or naive Bayes. The two-step approach also faces some limitations because of the tedious feature extraction process and it requires domain knowledge to design the features. Since the features are pre-defined, they cannot be easily generalized to new tasks.

Recently, deep learning methods are being widely used in many Natural Language Processing (NLP) related tasks and are improving the state-of-the-art of NLP [21] [6]. Such models attempt in an end-to-end manner to learn feature representations and perform classification.

The most important factor in improving the current deep learning methods like Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN) apart from efficiency and accuracy is the reduction in the dimension of inputs. We aim to generalize the task so that the model can be used on similar datasets. We attempt to achieve this by using a four-step strategy known as EEAP.

Our main aim is to recognize the terms as precisely as possible, so it is important to understand the context between the sequence of words. Embeddings like GloVe and word2vec ignore this information. Therefore, we have used BERT (Bidirectional Encoder Representations from Transformers) to capture the contextual information [4] that helps recognize our proposed hypothesis better.

The major contributions we like to mention here are; we have defined the traditional NLP task as a deep learning model which can be custom trained based on requirements. This model is effective in determining which sequence of words are terms compared to the statistical approach. We have also addressed the importance of contextual information in term recognition task in this tool by implementing BERT. Finally, in Section 5 we expose our results and show how our model outperformed the baseline model ATR4S referred in Section 2 by 28%. Our model also eliminates the need for multiple ranking and scoring algorithm to recognize terms in a given set of documents.

2 Related Work

Rule-based and statistical ATR researches

Rule-based and statistical ATR methods [13] focused on parts-of-speech (PoS) for multi-word constituents. Such work contributed to the recognition of words by pattern-based approaches such as linguistic filters. Each word is tagged with its associated PoS in the linguistic filter system, and the domain-specific term is defined based on the tag. A list of terms identified by the linguistic filters (linguistic process) is commonly referred to as "candidate terms" (CT).

Each sequence of words in the Candidate Terms (CT) (n-grams) is then given a score using statistical approaches. The score tells how likely the term is to be valid. The scores [13] are either the measures of "unithood", which attempts to identify if multi-word CT constituents form a collocation rather than a co-occurrence by chance; or the measures of "termhood" focus on measuring how likely a candidate term, CT, is a domain-specific concept. The most commonly used technique to score the CT is to consider "frequency of occurrence". The most recent term weighting scheme is TF-IDF which weights each term based on the number of occurrences within the document as well as within the entire corpora. These methods are used to filter the CT. Once filtered, because of their low ambiguity and high specificity, these extracted terms then can be used for many tasks including machine translation [5], information retrieval [15], ontology construction and ontology enrichment [2].

Baseline Model: ATR4S

Recent work on ATR is conducted by ART4S [1]. It comprises 13 state-of-the-art (SOTA) methods for ATR and implements the whole pipeline from text document pre-processing, to term candidate collection, term candidate scoring, and finally, term candidate ranking. The

text in the corpus is first split into sentences as part of pre-processing, tokenize and extract part-of-speech tags and lemmas for obtained tokens. Once the texts are pre-processed, the next step is "term candidate collection" – In this step, consecutive word n-grams (typically 1 to 4) of specified orders are extracted and three basic filters are applied (1. The noise filter: To remove the unnecessary tags like HTML tags. 2. Stop word filter and 3. PoS tagging). This gives a list of rare words. Words are then vectorized using the word2vec model. Each word in the list is scored using 13 SOTA methods (TF-TDF, C-values, etc.,). Once the scoring is done, the term is ranked to find how relevant a term is for being a key-term or valid term.

Other related works

JATE 2.0 [24] is also closely related to ART4S [1] and uses 10 state-of-the-art methods and is written in Java. Data is processed using traditional methods as in ART4S (pre-processing). The pre-processed data is then passed to "candidate extraction". JATE 2.0 uses Solr's analyzers for word vectorization which is a large text processing library. JATE 2.0 allows the user to customize the analyzer based on individual needs. The obtained candidates are then processed using different ATR algorithms which assigns the score and rank to the candidate terms.

AdaText [25] is another tool that is used in ATR. This tool improves on the TextRank algorithm to generate better performance. This provides generic methods to improve performance in any domain when coupled with an existing ATR method. AdaText uses GloVe word embeddings on the 2 datasets. The main limitation of AdaText [25] is the lack of understanding of the relation between the threshold used for selecting words on the TextRank.

All the works mentioned above provide some solution to identify domain-specific terms but often result in an error-prone system due to the use of context-free models like word2vec and GloVe. These models generate a single word embedding for each word in the CT, resulting in unidirectional language models. This limits the choice of architecture that can be used during pre-training [4]. Each candidate term needs to be evaluated not only based on the frequency of occurrence but also the context. This contextual information is often found on both the left-hand side and the right-hand side of the term. To address this issue a new approach is proposed here – using BERT (Bidirectional Encoder Representations from Transformers) embeddings.

Stanford University has recently used BERT in its ATR method for glossary terms [10]. The focus is on biology terms for the online textbook, Inquire. They have used the CNN along with BERT embedding to extract the terms for one domain (biology). The data was prepared manually, and it is a laborious process. The embeddings are generated only for unigram and hence the multi-word key-terms are ignored here.

So far, all the rule-based methods and tools available for ATR used context-free models and hence ignores the conceptual attribute for the term. Terms can be identified with more accuracy if the contextual property is considered. There are recent advances in using contextual models for term extraction [20] which uses BERT to fine-tune the terms extracted using feature-based approach. In contrast, we propose the idea of using BERT embedding which can capture the context of the given word and based on the context each candidate term can be ranked. Our hypothesis here is that the term classified as key-term by this process will be more accurate and reliable compared to other ATR tools.

3 Methodology

First, we use the Wikipedia titles as positive examples and generate random n-grams (of length 1–4) as a possible set of negative terms. We filter out most of the unrelated n-grams using Term Frequency - Inverse Document Frequency (TF-IDF), this ensures that we train the model on challenging negative terms instead of random noise. If these n-grams are not already present in our positive example, it is added as a weak negative example. Finally, this dataset is transferred to a CSV for training purpose.

The model consists of BERT for embedding, Bi-LSTM for encoding, Attention for reducing the input vector, ADAM optimizer [23] for training and a final prediction layer using a sigmoid output forming the EEAP structure.

3.1 BERT embeddings

BERT can be used to extract features like word and sentence embedding vectors from text data. These vectors are used as feature inputs to downstream NLP models like LSTM, GRU, etc., NLP models require numerical vectors as inputs. Previously, texts were either interpreted as uniquely indexed values (one-hot encoding) or more usefully as neural word embedding where vocabulary words are mapped against fixed-length embedding features resulting from models such as word2vec or Fasttext (does not consider the context within which the word appears). BERT improves over word2vec by generating the embedding based on the words around the text. This information is useful in ATR and hence, we have chosen BERT embeddings.

The output representations from the BERT encoding layer are summed element-wise to generate a single representation with shape (1, n, 768) for sequence embedding or (n,768) for word embedding.

3.2 Encode

Provided a sequence of word vectors, the encode step generates a matrix where each row represents the meaning of each token while paying attention to the context of the rest of the sentence.

Figure 1 Encode [8].

In this project, we have used a bidirectional LSTM. LSTM is a variant of RNN which is developed as a remedy to the problem of vanishing gradients and exploding gradients [7]. The key to solving the problem is by adding gates and a cell state to the RNN. A gate is a non-linear function (usually a sigmoid) followed by multiplication.

3.3 Attend

The attend step reduces the size of the matrix produced by the encode step to a single vector. In the process of reducing the matrix size, we lose most of the information. It is required to retain important information and hence the context vector is crucial. This vector tells which information to discard.

Figure 2 Attend [8].

We have employed an attention mechanism that learns the context vector as a parameter in the model. This is inspired by the recent research conducted by Harbin Institute of Technology [16] called "inner-attention". Instead of using the target sentence to attend words in the source sentence, inner-attention uses the sentence's previous-stage representation to attend to words that appeared. This approach results in a similar distribution of weights compared to other attention mechanisms and assigns more weight to important words. This approach produces precise and focused sentence representations for classification. Hence, the "inner-attention" is selected for this step. It is inspired by the concept of how human can roughly form a sense of which part of the sentence is important based on previous experiences. Mathematically, this mechanism can be written as follows:

$$M = \tanh(W^y Y + W^h R_{ave} \otimes e_L) \tag{1}$$

$$\alpha = softmax(w^{TM}) \tag{2}$$

$$R_{att} = Y\alpha^T \tag{3}$$

where, Y is a matrix of output vectors of bi-LSTM, R_{ave} is the output of mean pooling layer, e_L represents the bias matrix generated from the encoded input, α denotes the attention vector and R_{att} is the attention-weighted sentence representation. W^y and W^{TM} represents the attentive weight matrix.
This process makes the attention mechanism a pure reduction task, which can replace the sum or average pooling step.

Figure 3 Predict [8].

3.4 Predict

Once the input data is reduced to a single vector, we can learn the target representation in this step. Target representation may be a class label, a real value, a vector, etc. In our work, the target representation is a class label. 0 if the sequence of words is non-terms and 1 if the sequence of words contributes to being a term.

The predict layer is the last in our EEAP model. It receives the input from the attention layer, a 2D tensor, and the input is passed through a dense layer with a "sigmoid" activation function. Since we have to predict either 0 or 1, we have used the "sigmoid" function at the last layer of the model i.e., the prediction layer. This function converts any real value into another value in the range of 0 to 1. We map these predicted values to the probabilities of the CT being a term. If the probability is less than 0.5 then it is not a term, or if the probability is greater than 0.5 then it is classified as a term.

4 Experimental Settings

4.1 Data

There are two stages of data preparation for this model.

Stage 1 – Complete dataset preparation: Wikipedia titles are added to a list as positive examples and random n-grams are added as weak negative examples, this list is called candidate terms. If the generated n-gram is not in positive examples, then it is labelled as 0 (a negative term). All the Wikipedia terms (positive term) are labelled ad 1. This list of labelled data is converted into CSV to pass on to the next stage.

Stage 2 – Training and testing data preparation: The CSV file is loaded into the project. The data is then divided into train and test data in an 80:20 ratio. The text and label are separately loaded into the list from each train and test data. Text data is tokenized using BERT's FullTokenizer and padded to bring all input length to the same length. This data is then passed to the BERT layer and then to the EEAP model to make the prediction.

4.2 Model Architecture

The overall model architecture consists of several layers as explained below:

1. **Embedding layer:** This layer takes the BERT embedding matrix as input. The BERT embedding is of shape (n, 768), where n is the vocabulary size. Once the embedding matrix is passed through the embedding layer, the resulting output is a 3D tensor of shape (batch_size, max_len, embedding_dim) i.e., (?,64, 768) in our case. The batch size will be substituted at the run time.

2. **Encode layer:** A bidirectional LSTM layer is used as an encoding layer with 250 hidden units, dropout and recurrent dropout is set to 0.1 which will drop the fraction of the units for the linear transformation of the inputs and recurrent state respectively. The resulting output is of shape (batch_size, max_len, hidden_units) i.e., (?, 64, 20).

3. **Attention layer:** The attention layer takes the input from the encoding layer (3D tensor) and squeezes the input to 2D tensor and returns (batch_size, hidden_units) i.e., (?, 20). The main intention behind this step is input reduction by retaining only important information. The reduction is done using the tanh activation function. A dot product of the input matrix and weight along with the bias is passed to the activation function. The result of the tanh lies in-between -1 and 1. The benefit is that negative inputs are mapped highly negative and the zero inputs in the tanh graph are mapped near-zero thus helping to retain only important information. Attention is also explained in Section 3.3

4. **Feed Forward fully connected layer:** A dense layer is a fully connected neural network layer. We have specified 100 hidden units in the dense layer with the activation function Rectified Linear Unit (ReLU). The number of units denotes the output size. Activation in the dense layer sets the element-wise activation function to be used in the dense layer. We have used multiple dense layers in the model with the last layer being activated with the "sigmoid" activation function with 1 output node. Activation function selection is explained in Section 4.3

5. **Dropout layer:** The dropout layer randomly sets the specified fraction of input nodes to 0 at each stage during training which helps prevent over-fitting. In this project, we are using a single dropout layer with a 0.1 drop rate to avoid over-fitting. Figure 4 shows how the model begins with over-fitting the data and over multiple iterations, the model avoids over-fitting. This is achieved by the dropout layer. This value was selected as the best fit after running the model with different fractions.

4.3 Hyper-parameters

Optimizer

Optimizers are algorithms or techniques used to adjust the neural network's properties such as weights and learning rate to reduce the losses. Optimizers help in getting the results faster. We have used the Adam optimizer [12] [23] for building the EEAP structured model. Adam optimizer is an extension of stochastic gradient descent with adaptive learning rate methods to find individual learning rates for each parameter.

Loss Function

We have used the binary cross-entropy loss function as the problem we are trying to solve here is, the binary classification problem.

Activation function

The sigmoid activation function (also called the logistic function), is a very popular activation function for the neural network. The input to the function is transformed into a value between 0.0 and 1.0. Since ours is a binary classification problem, we have used this function in the last layer of the model to get the probability of the input being term, i.e., less than 0.5 is a non-term and greater than 0.5 is a term.

Learning rate

The learning rate is a tuning parameter in an optimization algorithm that determines the size of the step at each iteration while moving toward a minimum of a loss. Since it influences the extent to which newly acquired information outweighs old information, it represents the speed at which a machine learning model learns. We are setting the learning rate to 0.001 after running the model with different rates.

Decay/epsilon factor

Epsilon is the parameter used to avoid the divide by zero error when the gradient almost reaches zero. Setting epsilon to a very small value would result in larger weight updates and the optimizer becomes unstable. The bigger the value you set, the smaller the weights updates and the model training process becomes slow. Therefore, we have chosen 0.0001 as a good value for epsilon after running the model a few times with different values.

5 Results

Statistical Evaluation

The dataset used to train the model is Wikipedia titles as positive examples and random n-grams as weak negative examples. The model is then evaluated against 2 other datasets – GENIA [11] and Krapivin [14]. Table 1 gives the dataset description.

Table 1 Dataset description.

Dataset	Domain	Docs	Words (thousands)	Expected terms	Source of terms
GENIA	Bio medicine	2000	494	35,104	Authors' keywords
Krapivin	Computer science	2304	21	8766	Authors' keywords

The candidate terms were extracted using the TF-IDF method and compared against the expected terms from the datasets. Table 2 gives the candidate terms extracted across all the datasets.
This way of filtering candidate terms is useful while we pass the entire document to the model to predict the terms in it.

Table 2 Candidate terms.

Dataset	N-grams	Candidate terms	Candidates among expected terms
GENIA	10000	7341	2659
krapivin	10000	7370	4150

EEAP model performance evaluation

The deep learning model is trained to recognize the terms with a total of 1,291,921 training samples and 322,981 testing samples. The complete Wikipedia dataset consists of 1,614,902 samples with 1,314,902 positive examples and 300,000 negative examples.

We tested the model with different combination of hyper-parameters along with two selected encoders LSTM and GRU to decide which of these combinations results in better accuracy. The Food and Agriculture Organization (FAO) dataset is used for this evaluation. The FAO dataset is described in Table 3.

Table 3 FAO dataset description.

Domain	Agriculture
Docs	779
Words	26,672
Expected terms	1554
Source of terms	Author's keywords
Candidate terms	3895
Candidates among expected terms	862

We have used 0.001 as the learning rate since it is the standard learning rate set across the optimizer. Encoders have 250 hidden nodes for all iterations. To avoid lengthy iteration and due to resource constraints, we are considering the smaller dataset FAO for this comparison. Table 4 gives the model evaluation result.

Table 4 Model performance for different hyper-parameter combinations on FAO dataset.

Encoder	Optimizer	F1-score	Precision	Recall	Accuracy
GRU	Adam	0.0673	0.6296	0.0355	56.3%
GRU	SGD	0.0609	0.6183	0.0304	56.1%
GRU	Adadelta	0.0609	0.6183	0.0304	56.1%
GRU	RMSProp	0.073	0.653	0.0345	56.2%
LSTM	Adam	0.1947	0.8253	0.1104	60.5%
LSTM	SGD	0.0609	1.6183	0.0304	56.1%
LSTM	Adadelta	0.6093	0.4381	1.0	43.8%
LSTM	RMSProp	0.063	0.643	0.0335	55.2%

Along with the combination mentioned in Table 4, the loss function has also been changed to other loss functions like "categorical cross-entropy", "sparse categorical cross-entropy". Since this project is a binary classification, we are not moving further to use these loss functions as it does not fit our problem definition. We have evaluated the model performance with parameters that fit the project requirement and problem definition. After evaluating all the experimental results, with LSTM as encoder, Adam optimizer and binary cross-entropy loss function are selected as the best match for the model.

Figure 4 shows the model's training and validation accuracy over 100 epochs. We can see that the training accuracy keeps increasing over the iterations and this is because the model learns in each iteration. In the beginning, the validation accuracy is more than training accuracy which indicates over-fitting. Since we have used dropout layers in the model, the model avoids over-fitting over the iterations. At around 50 iterations, training accuracy crosses over validation accuracy. This indicates that the model is now learning for the training data efficiently.

Figure 5 shows the loss incurred over 100 epochs. The loss function intends to make the model learn. The loss is propagated back to the hidden nodes and the model learns to minimize these losses. Our model's loss keeps decreasing over the iterations and this shows that the model is learning better in each step. We further ran the model for 1000 iteration to find the convergence, Figure 6 shows the convergence.

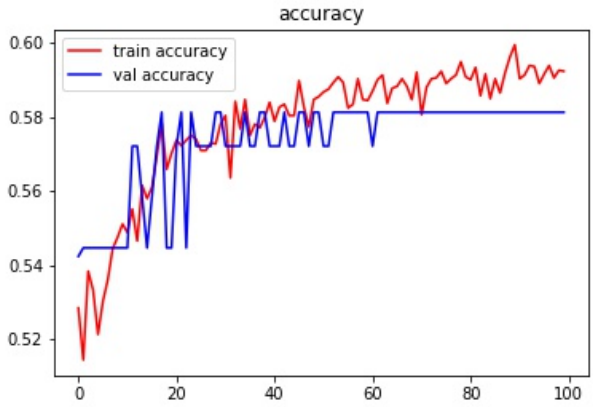


Figure 4 Model accuracy over 100 iteration.

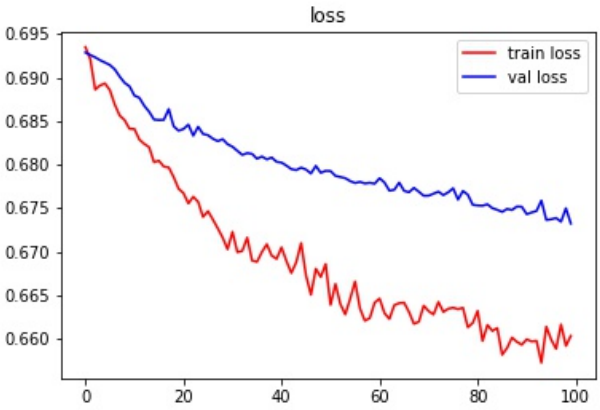


Figure 5 Decrease in loss over 100 iteration.

Evaluation on different dataset

The model is evaluated against two different datasets – GENIA and Krapivin as mentioned in Section 5. Table 5 shows the evaluation of these two datasets. The result is also evaluated against the base model ATR4S [1] and results are included in the Table 5. The FAO dataset used here is the held-out data to perform the evaluation.

Table 5 Evaluation on different datasets.

Comparison – EA-ATR(A) vs ATR4S(B)					EA-ATR model	
Dataset	A precision	B precision	A accuracy	B accuracy	F1-score	Recall
GENIA	0.8045	0.7760	60%	24%	0.7460	0.6955
Krapivin	0.6345	0.4279	62%	42%	0.7612	0.9511

(ATR4S model recall and F1-score not available for comparison)

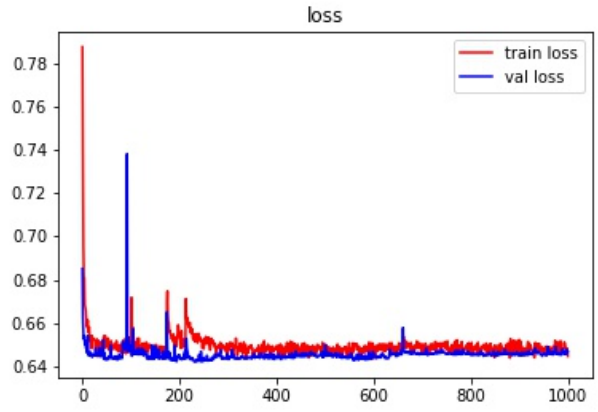


Figure 6 Convergence in loss over 1000 iteration.

Along with the precision, recall and accuracy metrics, we can extract the confusion matrix to evaluate the performance of the classifier. The idea is to count the number of times terms are classified as non-terms and vice-versa. Figure 7 shows the confusion matrix on evaluation dataset (FAO Terms).

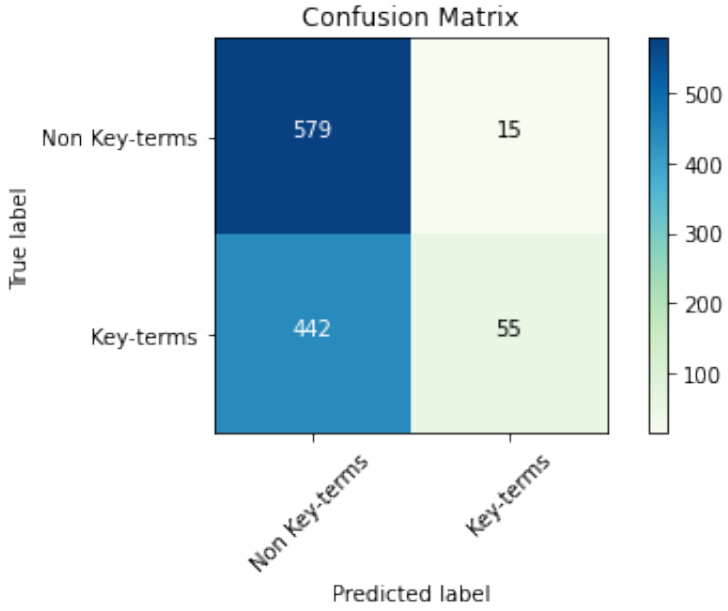


Figure 7 Confusion Matrix.

The model is well trained in predicting the non-terms. It is important to differentiate non-terms from terms because the ratio of non-terms in the document is more compared to terms. Although the model is a little biased towards non-terms, which is mainly because of the domain-specific dataset we are using, the model performs better considering the dataset used to train the model. This model stands as a new state-of-the-art for ATR using deep learning techniques. The model performs overall 28% better than the base model [1].

6 Conclusion

Current advances in NLP frameworks and applications focused on deep learning [22] have achieved better efficiency over many state-of-the-art NLP tasks, such as question answering and machine translation. This research is an attempt to show that deep learning models perform better and are more reliable than conventional automatic term recognition algorithms.

The model performs 28% better than the ATR4S [1] base model. The model also performs remarkably well on the GENIA and Kraplivin evaluation datasets. The simulations are a clear example of a deep learning model being applied to NLP tasks by reducing the repetitive computational requirement for each dataset and extracting automatic terms more precisely.

This method has the potential to be used as a multilingual model as it does not require any annotations. This is a future enhancement we would like to experiment with and see how well this works for different analytic and synthetic languages.

References

1 Nikita Astrakhantsev. ATR4S: toolkit with state-of-the-art automatic terms recognition methods in Scala. *Language Resources and Evaluation*, 52(3):853–872, 2018.

2 Nikita A Astrakhantsev, Denis G Fedorenko, and D Yu Turdakov. Methods for automatic term recognition in domain-specific text collections: A survey. *Programming and Computer Software*, 41(6):336–349, 2015.

3 James R Curran and Marc Moens. Improvements in automatic thesaurus extraction. In *Proceedings of the ACL-02 workshop on Unsupervised lexical acquisition*, pages 59–66, 2002.

4 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *arXiv preprint*, 2018. `arXiv:1810.04805`.

5 Éric Gaussier. Flow network models for word alignment and terminology extraction from bilingual corpora. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98/COLING '98, page 444–450, USA, 1998. Association for Computational Linguistics. `doi:10.3115/980845.980921`.

6 Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

7 Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

8 Matthew Honnibal. Embed, encode, attend, predict: The new deep learning formula for state-of-the-art NLP models. *Blog, Explosion, November*, 10, 2016.

9 Kyo Kageura and Bin Umino. Methods of automatic term recognition: A review. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 3(2):259–289, 1996.

10 Kush Khosla, Robbie Jones, and Nicholas Bowman. Featureless deep learning methods for automated key-term extraction, 2019. URL: `https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1194/reports/custom/15848334.pdf`.

11 J-D Kim, Tomoko Ohta, Yuka Tateisi, and Jun'ichi Tsujii. GENIA corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(suppl_1):i180–i182, 2003.

12 Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint*, 2014. `arXiv:1412.6980`.

13 Ioannis Korkontzelos, Ioannis P. Klapaftis, and Suresh Manandhar. Reviewing and evaluating automatic term recognition techniques. In Bengt Nordström and Aarne Ranta, editors, *Advances in Natural Language Processing*, pages 248–259, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

14 Mikalai Krapivin, Aliaksandr Autaeu, and Maurizio Marchese. Large dataset for keyphrases extraction. Technical report, University of Trento, 2009.

15 Yang Lingpeng, Ji Donghong, Zhou Guodong, and Nie Yu. Improving retrieval effectiveness by using key terms in top retrieved documents. In David E. Losada and Juan M. Fernández-Luna, editors, *Advances in Information Retrieval*, pages 169–184, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

16 Yang Liu, Chengjie Sun, Lei Lin, and Xiaolong Wang. Learning natural language inference using bidirectional LSTM model and inner-attention. *arXiv preprint*, 2016. arXiv:1605.09090.

17 Diana Maynard, Yaoyong Li, and Wim Peters. NLP techniques for term extraction and ontology population, 2008.

18 Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. Deep learning based text classification: A comprehensive review. *arXiv preprint*, 2020. arXiv:2004.03705.

19 Michael P Oakes and Chris D Paice. Term extraction for automatic abstracting. *D. Bourigault, C. Jacquemin, and MC. L'Homme, editors, Recent Advances in Computational Terminology*, 2:353–370, 2001.

20 Ayla Rigouts Terryn, Veronique Hoste, Patrick Drouin, and Els Lefever. TermEval 2020: Shared task on automatic term extraction using the annotated corpora for term extraction research (ACTER) dataset. In *Proceedings of the 6th International Workshop on Computational Terminology*, pages 85–94, Marseille, France, May 2020. European Language Resources Association. URL: https://www.aclweb.org/anthology/2020.computerm-1.12.

21 Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

22 Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing. *CoRR*, abs/1708.02709, 2017. arXiv:1708.02709.

23 Zijun Zhang. Improved Adam optimizer for deep neural networks. In *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*, pages 1–2. IEEE, 2018.

24 Ziqi Zhang, Jie Gao, and Fabio Ciravegna. JATE 2.0: Java automatic term extraction with apache Solr. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2262–2269, Portorož, Slovenia, 2016. European Language Resources Association (ELRA). URL: https://www.aclweb.org/anthology/L16-1359.

25 Ziqi Zhang, Johann Petrak, and Diana Maynard. Adapted textrank for term extraction: A generic method of improving automatic term extraction algorithms. *Procedia Computer Science*, 137:102–108, January 2018. doi:10.1016/j.procs.2018.09.010.

Universal Dependencies for Multilingual Open Information Extraction

Massinissa Atmani[1]
LIRMM, University of Montpellier, 860 rue de St Priest, 34095 Montpellier, France
Amaris Research Unit, 25 boulevard Eugène Deruelle, 69003 Lyon, France

Mathieu Lafourcade
LIRMM, University of Montpellier, 860 rue de St Priest 34095 Montpellier, France

Abstract

In this paper, we present our approach for Multilingual Open Information Extraction. Our sequence labeling based approach builds only on Universal Dependency representation to capture OpenIE's regularities and to perform Cross-lingual Multilingual OpenIE. We propose a new two-stage pipeline model for sequence labeling, that first identifies all the arguments of the relation and only then classifies them according to their most likely label. This paper also introduces a new benchmark evaluation for French. Experimental Evaluation shows that our approach achieves the best results in the available Benchmarks (English, French, Spanish and Portuguese).

2012 ACM Subject Classification Computing methodologies → Information extraction

Keywords and phrases Natural Language Processing, Information Extraction, Machine Learning

Digital Object Identifier 10.4230/OASIcs.LDK.2021.24

1 Introduction

Open Information Extraction (OpenIE) seeks to extract facts and events asserted by a sentence through a predicate-argument representation. [26] presents OpenIE as *"a novel extraction paradigm that facilitates domain-independent discovery of relations extracted from text and readily scales to the diversity and size of the Web corpus"*. Many downstream NLP tasks [15] had benefited from OpenIE such as multi-document question answering and [8], event schema induction[1] and word embedding generation [22].

Most of the OpenIE systems focus on English, with only few ones proposing multilingual OpenIE [24, 21]. In this paper, we present a supervised approach to perform multilingual OpenIE by exploiting only Universal Dependency. Like [21], our approach handles multilingual text without non-English training datasets. We also derive a new benchmark for French by following annotation guidelines of [13]. We introduce a model for sequence labeling, consisting of two sub-modules. The first module is a multi-task model that extracts the predicate-relation, then seeks to find all the arguments given the extracted predicate relation. The second module takes as input the extracted predicate and arguments, then assigns the most likely label to each potential argument such as subject, object, temporal argument or location argument. The reason for such a design, stems from the recent trends in neural dependency parsing [6], where they aim to find the unlabeled dependency structure (topology of the syntactic tree), and only then assign a label for each predicted arc of the tree. More specifically, their model calculates the probability of an arc between each pair of words as well as a syntactic function label for each arc. In contrast to their approach, we only compute the probability between a word and the span of words representing the predicate phrase extracted in the previous step. In our setting, the predicted arcs indicate the extracted

[1] corresponding author

Table 1 OIE extractions example.

Sentence	Bennett confirmed when he addressed the Township Council tonight that the United States attorney's office had requested information from the township.
Extractions	(**A0:**Bennett**; P:**confirmed**; A1:**the United States attorney 's office had requested information from the township) (**A0:**the United States attorney 's office**; P:**had requested information from**; A1:**the township) (**A0:**he**; P:**addressed**; A1:**the Township Council**; A2:**tonight) (**A0:**Bennett**; P:**addressed**; A1:**the Township Council**; A2:**tonight)

arguments of the predicate phrase, those extracted arguments will be classified in the next stage. Our approach achieves the best results in all the languages against the existing systems (multilingual and non-multilingual). Finally, we show through the experiments results that current BERT-based approaches are not cross-domain friendly and fail when dealing with out-of-domain samples. We find that it is important to report this finding as domain-adaptation is the most important characteristic of OpenIE paradigm.

2 Related Work

2.1 Legacy systems

[16] classified rule-based OpenIE systems to three major approaches, according to the type of features exploited: shallow OpenIE, OpenIE via dependency parsing, and OpenIE via semantic parsing. Early OpenIE systems exploited only shallow syntactic parsing such as part-of-speech tagging and chunking [26, 7]. More advanced systems greatly enhanced performance by exploiting more advanced linguistic processing. [4] used dependency parse tree to decompose complex sentences into a set of independent clauses, where each type of a clause can express a relation with a predefined predicate-argument structure. Semantic Role Labeling (SRL) consists into labeling words of a sentence into their semantic role, such as agent, theme and instrument. The SRL task is somewhat similar to OpenIE task, and on account of the resource availability, [3] used a SRL parser to derive their system SRLIE. Several OpenIE systems extract relations mediated by verb predicate and ignore nominal relations [25] proposed RENOUN to extract nominal-based relations. [19] designed an OpenIE system tailored to relations expressed by demonyms and relational compound nouns. OPENIE4 was derived by merging SRLIE [3] and RelNoun [19] systems. They augmented OpenIE4 with an OIE system tailored to numerical relations as well as with a system to break conjunctions to derive OpenIE5.

2.2 Neural based systems

With the hype surrounding deep learning and language models neural methods have been employed for OpenIE task to bypass error accumulation in rule-based systems, with a focus on automatically deriving corpora large enough to train neural open information extraction systems. The obtained datasets are large enough to train deep learning models, but at the cost of being very noisy and erroneous. Hence, [12] proposed a Score and Filter framework to reduce redundancy and noise in those bootstrapped datasets. [23] addressed OIE as a sequence lebeling problem with the BIO (Beginning, Inside, Outside) template, using a Bi-LSTM with Softmax to each word of the sentence. [5] formulated OpenIE as a relation generation

problem, with an encoder-decoder architecture using attention mechanism. Inspired from recent work in SRL [18], [27] formulated OIE as a span selection problem, where they build two sequential modules, a former one predicting the predicate boundary with the encoded sentence as input, the latter one predicting arguments boundary with the predicate boundary and encoded sentence as input. [12] used a BERT encoder and an iterative decoder to keep track of the predicted extractions and to model their inter-dependencies. [11] addressed the OpenIE task as an iterative 2-D Grid Labeling task using a BERT encoder, such an approach helps to model dependencies between extractions while being much faster than [12]. They also augmented their model with a coordination analyzer to better deal with complex coordination structures.

2.3 Multilingual systems

Most OpenIE systems for languages other than English are ad-hoc and rule-based approaches, with limited performance. Among these approaches, two systems stand out: ArgOIE and PredPatt. [9] presented ArgOIE which takes as input the dependency parsing in CoNLL-X format, identifies the argument structures in the dependency analysis and extracts a basic set of propositions from each argument structure. ArgOIE supports OpenIE in four languages: English, Spanish, Portuguese and Galician. Similar to ArgOIE, PredPatt [24] also takes Universal Dependency [17] parse as input and returns a set of predicate-arguments structures by applying language-agnostic patterns. [21] proposed Multi2OIE, a sequence labeling model for OpenIE, which first predicts all relation arguments using BERT, then predicts the subject and object arguments associated with each relation using multi-head attention blocks. More precisely, it uses the multilingual version of BERT in order to support OpenIE in all the languages supported by BERT-Multilingual. Their approach supports multilingual text without non-English datasets, as their model is only trained on a corpus of English sentences.

3 Methodology

We introduce our proposed method in detail in this section. First, we give the task formulation and the overview of our approach to neural OIE in Section 3.1 and Section 3.2. Finally, we present the input representation and our model architecture for OpenIE respectively in Section 3.5 & Sections 3.3 and 3.4.

3.1 Problem Definition

Given a sentence $S = (w_1, w_2, ..., w_n)$, we first derive the dependency syntactic tree to obtain the POS tags and dependency relation embedding. We feed those embeddings to the model to produce a sequence tag $T = (y_1, y_2, ..., y_n)$, with the set of tags $Y = \{A0, P, A1, A2, O\}$. The produced sequence represents the tuple ($A0$:subject, P :predicate, $A1$:object, ...) in the BIES template format (Begin, Inside, End, Single).

Table 2 Example sentences and respective Open IE extractions.

OpenIE encoding example	
Sentence	Brady attempts to phone the sheriff .
Sequence labels	$A0_S$ P_B P_I P_E $A1_B$ $A1_E$ O
Output encoding	Brady_{A0_S} attempts_{P_B} to_{P_I} phone_{P_E} the_{A1_B} sheriff_{A1_E} $.O$
Tuple	($A0$:Brady, P :attempts to phone, $A1$:the sheriff)

3.2 Approach Overview

Following [23], we approach OpenIE task as a Sequence Labeling Problem with the BIES template (Begin, Inside, End, Single). Sequence Labeling aims to assign each word of the sentence its most-likely tag, producing a sequence tag $T = (y_1, y_2, ..., y_n)$. For each sentence, we extract one relation at one time, by considering at each iteration a candidate predicate word, from which we infer a binary mask $M = (m_1, m_2, ..., m_n)$. Our proposed model consists of two weakly bounded modules, the former one handles the predicate and argument inference, feeds the inferred predicate boundary to the latter, which classifies the extracted arguments.

3.3 Predicate-Argument Extractor

We follow the recent trends in neural dependency parsing [6],where the unlabeled dependency structure (topology of the syntactic tree) is extracted and only then the edges of the tree are assigned a label for. Our first sub-module aims at extracting the predicate-argument representation where the arguments are non-typed. Hence, the sub-module is optimized with regard to two tasks: predicate extraction and argument extraction and shares the same parameters for the two tasks, the later task depends on the output of the former task. The inputs for the sub-module are the concatenation of the three features: $E_{pos}, E_{dep}, E_{mask}$. The first feature is the part-of speech embedding, the second is dependency label embedding, and the third is the embedding of the binary predicate mask. Since we extract one relation at one time, E_{mask} is a simple binary vector to indicate which word of the sentence is the candidate predicate. The sub-module shares a Bi-LSTM layer for both tasks and exploits a CRF layer for each task. Given an input instance (S, M) with S a sentence and M a binary vector (0 and 1), for every word $w_i \in S$ we compute a feature vector:

$$x_i = E_{pos}(w_i) \oplus E_{dep}(w_i) \oplus E_{mask}(w_i) \tag{1}$$

The feature vector in 1 is fed to the Bi-LSTM, which computes a forward and backward hidden state vector:

$$v_i^{\rightarrow}, v_i^{\leftarrow} = BiLSTM(x_i) \tag{2}$$

then the forward and backward output of Bi-LSTM are averaged, and fed to a dense layer:

$$u_i = AVG(v_i^{\rightarrow}, v_i^{\leftarrow}) \tag{3}$$

$$h_i = Wu_i + b \tag{4}$$

Then, the representation is fed to the decoder of each task. Since both tasks use the same CRF decoder, we first introduce the CRF decoder.

3.3.1 CRF Decoder

Given the decoder's input sequence $H = \{h_i\}_{i=1}^n$ and a sequence of labels $Y = \{y_i\}_{i=1}^n$, the decoder computes the decoding score $S(H, Y)$.

$$S(H, Y) = \sum_{i=1}^{n-1} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} H_{i, y_i} \tag{5}$$

H is an $n \times k$ emission matrix, where n is the length of the sequence, k the number of distinct tags, and H_{ij} is the score of j-th tag at position i of the sequence. A is a $k \times k$ transition matrix, where A_{ij} represents the transition score from the i-th tag to the j-th tag.

Then $p(Y|H)$ is computed, a conditional probability over all possible tag sequences Y using Softmax, where Y_H represents possible tag sequences for H.

$$p(Y|H) = \frac{e^{S(H,Y)}}{\sum_{Y' \in Y_H} e^{S(H,Y')}} \tag{6}$$

While decoding, we search for the sequence having the maximum score y^*, which is done using the Viterbi algorithm.

$$y^* = argmax_{Y \in Y_H} S(H, Y) \tag{7}$$

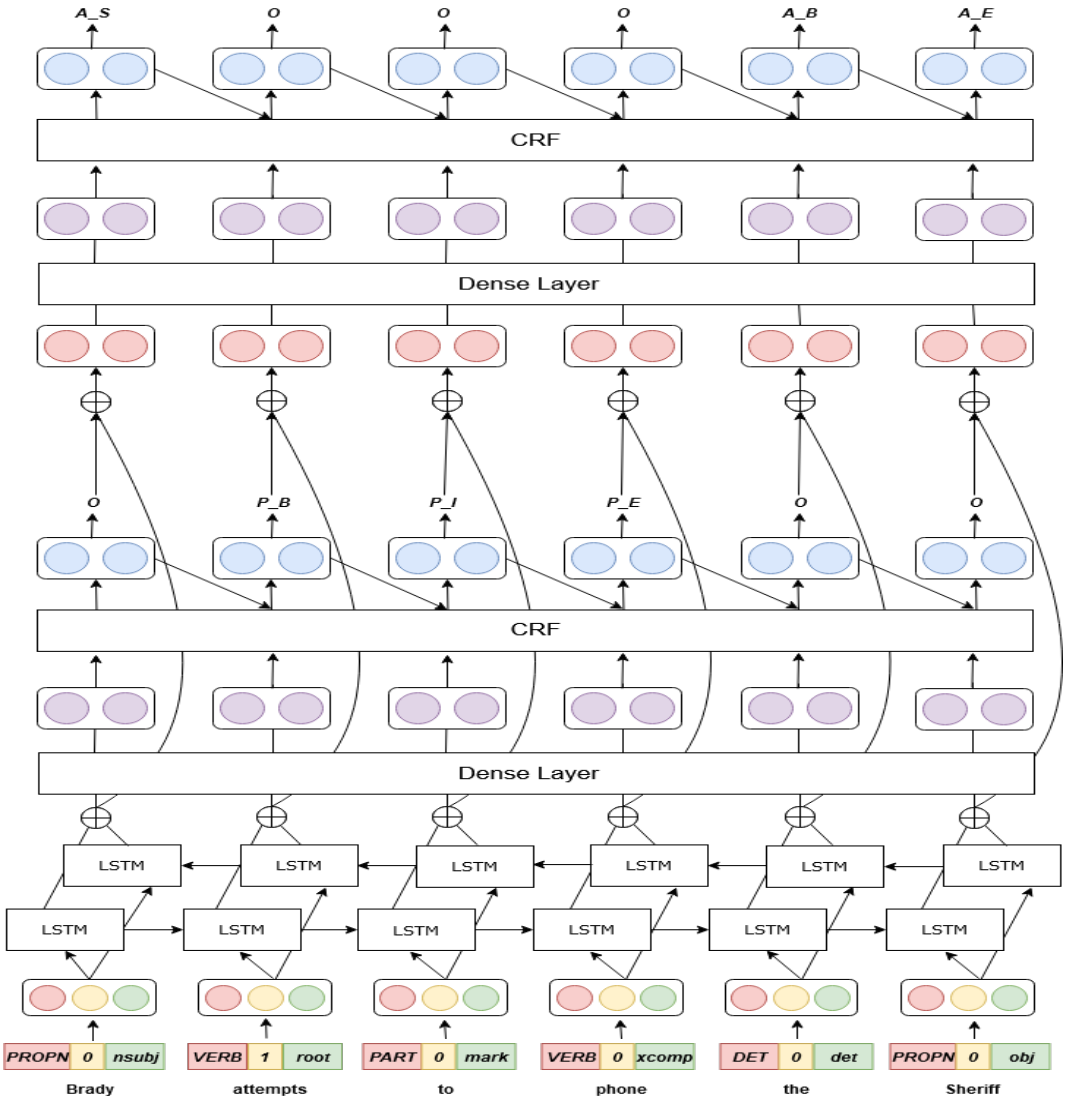


Figure 1 Architecture of the predicate-argument extractor.

The encoder output is first fed to the predicate extractor, that identifies the predicate phrase. After Extracting the predicate Equation (7), the predicate phrase is fed to the argument extractor as a binary vector that indicates the boundary of the extracted predicate. Finally,

the encoder output is concatenated with the output of the predicate task and is fed to the CRF decoder of the arguments extractor. The new representation is given by the following equation:

$$h_i(Argument) = h_i \oplus y_i(Predicate) \tag{8}$$

Both tasks are optimized jointly, and we maximize the log-likelihood of the correct tag sequence of each task on the training set $\{(H_j, Y_j)\}$, by minimizing the loss: the Negative Log Likelihood (NLL).

$$NLL = -\sum_j \log p(Y|H) \tag{9}$$

The loss of the sub-module is simply the sum of the loss of each task:

$$NLL = -\sum_j \log p(Y|H)_{predicate} - \sum_j \log p(Y|H)_{argument} \tag{10}$$

3.4 Argument Classifier

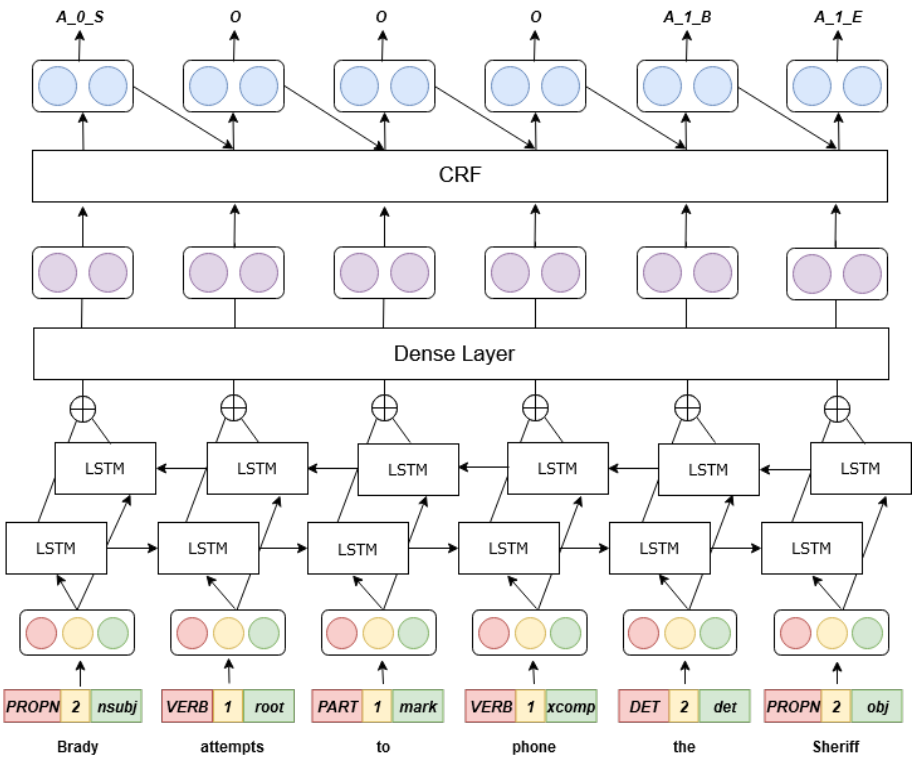


Figure 2 Architecture of the argument classifier.

After the predicate-argument inference, the first sub-module feeds the extracted predicate and arguments to the second sub-module. In addition to the part-of-speech and dependency label embedding, it takes as input another vector, that indicates for each word of the sentence if it: is part of the predicate phrase, is an argument of the extracted predicate or is none

of them. The model's input is the feature vector defined in Equation (11), and consists of the concatenation of the part-of-speech embedding, the dependency label embedding, and E_{pr-arg}, the vector inferred in the first stage that represents the extracted predicate and arguments.

$$x_i = E_{pos}(w_i) \oplus E_{dep}(w_i) \oplus E_{pr-arg}(w_i) \tag{11}$$

The sub-module exploits the same architecture as the first sub-module that consists of a CRF decoder stacked over a BiLSTM layer and seeks to assign the most likely label to the arguments extracted during the first stage. Like the predicate-argument extractor, the model is optimized during training by minimizing the negative log likelihood.

3.5 Input Pre-processing

We use the Stanza library [20] to obtain the part-of-speech tag and dependency parsing tree with the Universal Dependency representation [17]. For some POS categories such as pronouns and determinant, we add morphological information. The final POS vocabulary size consists of 31 categories, while dependency labels vocabulary size consists of 62 categories. Both part-of-speech and dependency labels embedding are encoded as one-hot encoding where each category is mapped to a different vector.

3.6 Confidence Score

As most OpenIE systems provide a confidence score for their extracted relations, which can be further exploited by downstream application to filter out relations. We use the Viterbi score Equation (7) of the argument classifier module as the confidence score of our model.

4 Experiments

In this section, the training datasets and hyperparameters are respectively presented in Section 4.1 and Section 4.2, then Section 4.3 and Section 4.4 describe the evaluation strategy and the evaluation benchmark. We present the ablation study and the baselines in Sections 4.5 and 4.6. We conclude the experiments study by a speed performance analysis in Section 4.7.

4.1 Dataset

In contrast to previous works, we pick manually annotated datasets used in [4] as our training data. Since those datasets contain binary relations, we re-annotate them to convert the binary-relations to n-ary relations. The annotation follows guideline of [13], except for the Anaphora resolution. Table 3 describes the datasets after re-annotation.

Table 3 Training Datasets.

Dataset	#Sentences	#Relations
Reverb	500	1,551
New York Times	200	642
Wikipedia	200	568

4.2 Hyperparameters

The Table 4 below, resumes the hyperparameters of our model, which are the same for both sub-modules. We trained our model using the Adam optimizer. After training, we validate our model on a validation dataset which was annotated by experts in [2] and consists of 50 sentences and 173 relations. The model's best performance on the validation dataset is reported in Table 5.

Table 4 Hyperparameters.

Model Hyper-Parameters	
LSTM Hidden size	128
LSTM Recurrent State dropout	0.3
LSTM Input dropout	0.3
LSTM Output dropout	0.3
Embedding dropout	0.1
Dense layer dropout	0.3
L2 Regularization	0.001
Embedding size	20
Batch size	5
Learning rate	0.001
Number of Hyper-Parameters	
Predicate-Argument Extractor	590,553
Argument Classifier	592,911
Full Model	1,183,464

Table 5 Evaluation Results on the validation benchmark.

System	CARB		CARB(1-1)	
	F1	AUC	F1	AUC
UD2OIE	**72.2**	**52.3**	**64.3**	**42.6**

4.3 Evaluation Strategy

We use the standard CARB [2] evaluation strategy to evaluate our system and the baselines. Following [11], we also report results for the CARB(1-1) scoring function, which penalizes incorrect splitting of coordination structures. We report the F1 score and the AUC (Area Under the Curve). Our model and the baselines are evaluated by exploiting the code and data used by [11] in their work.

4.4 Benchmark

In order to evaluate Multilingual OpenIE systems on Spanish and Portuguese, [21] derived Re-OIE2016_Sp and Re-OIE2016_Pt benchmarks by translating the English benchmark Re-OIE2016. We use these two benchmarks to evaluate the different systems on Spanish and Portuguese. Due to the lack of benchmark for French, we also annotate a benchmark by taking sentences from newspaper articles in the domain of finance, and which were described in [10]. To annotate the corpus, we follow the annotation recommendations of [13], which

were also followed by [2] to build CARB. The final evaluation benchmark consists of 506 sentences and 1,783 relationships. We use the standard CARB benchmark to evaluate the OpenIE systems on English.

Table 6 Evaluation Benchmark.

Dataset	#Sentences	#Relations
CARB	641	2,715
Re-OIE2016_Sp	595	1,508
Re-OIE2016_Pt	595	1,508
Finance_French	506	1,783

4.5 Ablation Study

We apply an ablation study to investigate the impact of our new architecture, which aims to separate the identification and labeling of the arguments. Hence, we consider a strong baseline slightly similar to the architecture used by [27]. Our proposed architecture introduces an auxiliary stage to identify the arguments of the extracted predicate before labeling those extracted arguments, while [27] identifies and labels the arguments of the extracted predicate simultaneously.

4.6 Baselines

We refer to our model as UD2OIE, while we refer to the baseline defined in the ablation study section as UD2OIE(-Arg Identification). For the English evaluation on the CARB benchmark, we pick rule-based, neural sequence labeling, and neural relation generation approaches. We choose ClauseIE [4], OpenIE4 [3], and OpenIE5 [3] as the rule-based baselines. As for sequence labeling baselines, we pick RnnOIE [23], SpanOIE[27], and OpenIE6 [11]. And the chosen baselines for relation generation approaches are NeuralOIE [5] and IMOJIE [12]. Finally, for the Multilingual Evaluation, we choose the two rule-based approaches PredPatt [24] and ArgOIE [9], while the only available neural baseline is Multi2OIE [21].

4.7 Speed performance

Since OpenIE systems must scale to the diversity and size of the Web corpus, we also report the inference time of our model on a batch of 3200 sentences (8477 relations) [23], which was also used in [11] to report the speed of the different systems. In contrast to [11] that reported the speed performance of the neural baselines using a V100 GPU, we report the speed of our model using 4 cores of Intel Core i5-8300H CPU. The speed performance of non-neural systems was reported in [11] using 4 cores of Intel Xeon CPU. We report the speed of our model with and without the execution time of the dependency parser.

5 Results and Analysis

This section discusses the key finding of the experiment results in Sections 5.1 and 5.2. The ablation study and domain adaptation results are discussed in Sections 5.3 and 5.4. Finally, the run-time analysis is reported in Section 5.5, and Section 5.6 provides an error analysis of the model. Table 7 shows multilingual extraction examples of our model.

Table 7 Extraction examples from UD2OIE for each language.

Sentence	Returning home, Ballard delivers her report, which her superiors refuse to believe.
English	(**A0:**Ballard**; P:**Returning**; A1:**home) (**A0:**Ballard**; P:**delivers**; A1:**her report) (**A0:**her superiors**; P:**refuse to believe**; A1:**her report)
Sentence	De retour chez elle, Ballard livre son rapport, que ses supérieurs refusent de croire.
French	(**A0:**Ballard**; P:**De retour chez**; A1:**elle) (**A0:**Ballard**; P:**livre**; A1:**son rapport) (**A0:**ses supérieurs**; P:**refusent de croire**; A1:**son rapport)
Sentence	Al volver a casa, Ballard entrega su informe, que sus superiores se niegan a creer.
Spanish	(**A0:**Ballard**; P:**volver a**; A1:**casa) (**A0:**Ballard**; P:**entrega**; A1:**su informe) (**A0:**sus superiores**; P:**niegan a creer**; A1:**su informe)
Sentence	Voltando para casa, Ballard entrega seu relatório, que seus superiores se recusam a acreditar.
Portuguese	(**A0:**Ballard**; P:**Voltando para**; A1:**casa) (**A0:**Ballard**; P:**entrega**; A1:**seu relatório) (**A0:**seus superiores**; P:**se recusam a acreditar**; A1:**seu relatório)

5.1 Monolingual Performance Results

The performance results for each system on the English CARB benchmark with the presented metrics are reported in the Table 8. The evaluation results show that our proposed method outperforms by a large gain the other systems.

Table 8 Evaluation Results of English OpenIE systems against the standard CARB benchmark.

System	CARB		CARB(1-1)	
	F1	AUC	F1	AUC
ClauseIE	45.0	22.0	40.2	17.7
OpenIE4	51.5	29.1	40.4	19.7
OpenIE5	46.7	24.5	41.2	19.6
SpanOIE	48.5	-	37.9	-
NeuralOIE	51.6	32.8	38.7	19.8
RnnOIE	49.0	26.0	39.5	18.3
IMOJIE	53.5	33.3	41.4	22.2
OpenIE6	52.7	33.7	46.4	26.8
UD2OIE	**58.2**	**39.0**	**49.9**	**29.7**

5.2 Multilingual Performance Results

The multilingual performance results for each system on the four benchmark using the CARB evaluation strategy are reported in the Table 9. The evaluation results show that our proposed method outperforms all the Multilingual OpenIE systems in all the benchmarks.

Table 9 Evaluation Results of Multilingual OpenIE systems against the different benchmarks.

System	English		French		Spanish		Portuguese	
	F1	AUC	F1	AUC	F1	AUC	F1	AUC
ArgOIE [9]	36.4	24.4	-	-	39.4	28.3	38.3	26.4
PredPatt [24]	44.6	34.6	42.0	34.7	44.3	39.8	42.9	38.0
Multi2OIE [21]	52.1	31.5	43.2	24.5	61.5	43.2	61.2	42.1
UD2OIE	**58.2**	**39.0**	**67.3**	**49.6**	**68.1**	**51.9**	**68.0**	**51.6**

5.3 Ablation Study Results

Table 10 Ablation study results.

System	English		French		Spanish		Portuguese	
	F1	AUC	F1	AUC	F1	AUC	F1	AUC
UD2OIE	**58.2**	**39.0**	**67.3**	**49.6**	**68.1**	**51.9**	**68.0**	**51.6**
UD2OIE (-Arg Identification)	57.0	35.6	64.5	44.9	64.3	43.7	64.9	45.4
System	English		French		Spanish		Portuguese	
	PRE	REC	PRE	REC	PRE	REC	PRE	REC
UD2OIE	61.4	**55.3**	72.7	**62.7**	**72.6**	**64.1**	72.8	**63.8**
UD2OIE (-Arg Identification)	**63.2**	52.0	**76.1**	56.0	71.0	58.7	**76.7**	56.2

The ablation's results resumed in Table 10, show that our proposed architecture provides a performance gain in all the benchmarks. Our proposed architecture targets the recall performance, it enhances the recall performance while resulting in a performance drop in the precision. We attribute this to the fact that searching all the relevant arguments before labeling them in the next stage is less complex and results in a more important number of predicate-argument relations. Hence, the recall performance increases as the number of predicate-argument relations increase. However, more erroneous predicate-argument relations will be propagated to the classifier module, which only seeks to label the extracted arguments and can't discard the erroneous ones, resulting in a performance drop in the precision.

5.4 Domain adaptation

While outperforming all the rule-based systems by a large margin on the English, Spanish and Portuguese benchmarks, Multi2OIE [21] only slightly outperforms PredPatt [24] on the French benchmark. To investigate the source of this pitfall, we derive a second French benchmark from the Wikipedia domain. To do so, we translate the English Wikipedia training dataset described in Table 3 to French, and manually annotate it following the the same guideline [13]. The Table 11 results show that Multi2OIE outperforms PredPatt by a

Table 11 Evaluation Results against the French version of Wikipedia benchmark.

System	French	
	F1	AUC
PredPatt [24]	37.6	30.4
Multi2OIE [21]	**53.6**	**32.9**

large margin on the French Wikipedia benchmark. We conjecture that Multi2OIE, which is based on BERT, achieves good performance on the Wikipedia benchmark only because BERT was pre-trained on Wikipedia data. Also because of BERT, Multi2OIE is unstable and fails when facing out-of-domain samples like financial texts. As reported by [14], despite their ability to extract language agnostic representations in their multilingual version, language models such as BERT only capture domain specific features and do not extract domain invariant features. Hence, BERT based approaches such as Multi2OIE are not cross-domain friendly, which violates the OpenIE paradigm principle.

5.5 Runtime Analysis

The Table 12 shows that our model can process approximately 20 sentences by second, despite being run on CPU and not on GPU. It also shows that our model can process 141.2 sentences by seconds if we exclude the dependency parsing run-time. While the reported results are not fair because of performance gap between CPU and GPU, the Table 12 shows that our proposed model achieves comparable results with the fastest rule-based approach (uses a semantic parser) on CPU.

Table 12 Performance Speed of OIE systems.

System	Speed Sentences/Second
ClauseIE	4.0
OpenIE4	20.1
OpenIE5	3.1
SpanOIE	19.4
NeuralOIE	11.5
RnnOIE	**149.0**
IMOJIE	2.6
OpenIE6	31.7
UD2OIE	20.1
UD2OIE (W/o Stanza)	141.2

5.6 Errors Analysis

As expected, the main source of errors was due to propagation errors of the parser. We find that our system fails at complex linguistic constructions. The last example in Table 13 shows an example of gapping, a type of ellipsis, where our system fails at extracting the corresponding relations. The Stanza library we used, regards the gapping as a simple conjunction clause, and feeds an incorrect syntactic tree to our model. Another important source of error was the n-ary argument field, where the n-ary relation was extracted as a binary relation, with the n-ary argument either missing or being in the object field. The first example in Table 13 shows an example due to ambiguity of preposition attachment, where *the Battle of Jamal* is extracted as part of the object field *Ali's army*. Also, our system fails more often at extracting nominal-based relations, as shown in Table 13. Finally, the last example in Table 13 shows a language-specific construction specific to French (*agentive indirect object* (expressed by **iobj:agent** in the UD syntactic tree) where the initial agent (the pronoun *lui* in the example) has been demoted and became an indirect object. Since our system was trained on English data, it will naturally fail when facing these language-specific constructions.

Table 13 Error Types.

Error Type	Example
N-ary arguments	And he was in Ali's army in the Battle of Jamal. **Extracted:** (**A0:**he**; P:**was in**; A1:**Ali's army in the Battle of Jamal) **Gold:** (**A0:**he**; P:**was in**; A1:**Ali's army**; A2:**in the Battle of Jamal)
Nominal	FBI Director Clive Anderson is the same kind of avuncular superior as Chief Brandon. (**A0:**Clive Anderson; **P:**[be] Director [of]; **A1:**FBI)
Complex linguistic constructions	A cafeteria is also located on the sixth floor , a chapel on the 14th floor , and a study hall on the 15th floor. **Extracted:** (**A0:**A cafeteria**; P:**is also located on**; A1:**the sixth floor) **Extracted:** (**A0:**A cafeteria**; P:**is also located on**; A1:**a chapel on the 14th floor) **Extracted:** (**A0:**A cafeteria**; P:**is also located on**; A1:**a study hall on the 15th floor) **Gold:** (**A0:**A cafeteria**; P:**is also located on**; A1:**the sixth floor) **Gold:** (**A0:**a chapel**; P:**is also located on**; A1:**the 14th floor) **Gold:** (**A0:**a study hall**; P:**is also located on**; A1:**the 15th floor)
Language-Specific constructions	Google et Facebook en embuscade face à Apple, seul Google lui tient un peu tête. Google and Facebook in ambush against Apple, only Google is standing up to it a bit. **Extracted:** (**A0:**Google**; P:**tient un peu tête**;**) **Gold:** (**A0:**Google**; P:**tient un peu tête**; A1:**lui)

6 Conclusion and Future Work

In this work, we proposed an approach for multilingual OpenIE, while introducing a new benchmark for French. We showed that our approach adapts to other languages without training data of the target language. We introduced a simple but effective model, that outperforms the standard two steps-based approaches (extract predicate then arguments). The experiment findings suggest that current BERT-based approaches are not cross-domain friendly and do not support domain adaptation [14].

References

1 Niranjan Balasubramanian, Stephen Soderland, Oren Etzioni, et al. Generating coherent event schemas at scale. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1721–1731, 2013.

2 Sangnie Bhardwaj, Samarth Aggarwal, and Mausam Mausam. CaRB: A crowdsourced benchmark for open IE. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6262–6267, Hong Kong, China, November 2019. Association for Computational Linguistics. doi:10.18653/v1/D19-1651.

3 Janara Christensen, Mausam, Stephen Soderland, and Oren Etzioni. Semantic role labeling for open information extraction. In *Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading*, pages 52–60, Los Angeles, California, June 2010. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/W10-0907.

4 Lucianno Del Corro and Rainer Gemulla. Clausie: clause-based open information extraction. In *Proceedings of the 22nd international conference on World Wide Web*, pages 355–366, 2013.

5 Lei Cui, Furu Wei, and Ming Zhou. Neural open information extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 407–413, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi:10.18653/v1/P18-2065.

6 Timothy Dozat and Christopher D Manning. Deep biaffine attention for neural dependency parsing. *arXiv preprint*, 2016. arXiv:1611.01734.

7 Anthony Fader, Stephen Soderland, and Oren Etzioni. Identifying relations for open information extraction. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545, Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/D11-1142.

8 Angela Fan, Claire Gardent, Chloé Braud, and Antoine Bordes. Using local knowledge graph construction to scale seq2seq models to multi-document inputs. *arXiv preprint*, 2019. arXiv:1910.08435.

9 Pablo Gamallo and Marcos Garcia. Multilingual open information extraction. In *Portuguese Conference on Artificial Intelligence*, pages 711–722. Springer, 2015.

10 Ali Jabbari, Olivier Sauvage, Hamada Zeine, and Hamza Chergui. A French corpus and annotation schema for named entity recognition and relation extraction of financial news. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2293–2299, Marseille, France, May 2020. European Language Resources Association. URL: https://www.aclweb.org/anthology/2020.lrec-1.279.

11 Keshav Kolluru, Vaibhav Adlakha, Samarth Aggarwal, Mausam, and Soumen Chakrabarti. OpenIE6: Iterative Grid Labeling and Coordination Analysis for Open Information Extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3748–3761, Online, November 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-main.306.

12 Keshav Kolluru, Samarth Aggarwal, Vipul Rathore, Mausam, and Soumen Chakrabarti. IMoJIE: Iterative memory-based joint open information extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5871–5886, Online, July 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.521.

13 William Lechelle, Fabrizio Gotti, and Phillippe Langlais. WiRe57 : A fine-grained benchmark for open information extraction. In *Proceedings of the 13th Linguistic Annotation Workshop*, pages 6–15, Florence, Italy, August 2019. Association for Computational Linguistics. doi:10.18653/v1/W19-4002.

14 Juntao Li, Ruidan He, Hai Ye, Hwee Tou Ng, Lidong Bing, and Rui Yan. Unsupervised domain adaptation of a pretrained cross-lingual language model. *arXiv preprint*, 2020. arXiv:2011.11499.

15 Mausam Mausam. Open information extraction systems and downstream applications. In *Proceedings of the twenty-fifth international joint conference on artificial intelligence*, pages 4074–4077, 2016.

16 Filipe Mesquita, Jordan Schmidek, and Denilson Barbosa. Effectiveness and efficiency of open relation extraction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 447–457, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/D13-1043.

17 Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL: `https://www.aclweb.org/anthology/L16-1262`.

18 Hiroki Ouchi, Hiroyuki Shindo, and Yuji Matsumoto. A span selection model for semantic role labeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1630–1642, Brussels, Belgium, October–November 2018. Association for Computational Linguistics. `doi:10.18653/v1/D18-1191`.

19 Harinder Pal and Mausam. Demonyms and compound relational nouns in nominal open IE. In *Proceedings of the 5th Workshop on Automated Knowledge Base Construction*, pages 35–39, San Diego, CA, June 2016. Association for Computational Linguistics. `doi:10.18653/v1/W16-1307`.

20 Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online, July 2020. Association for Computational Linguistics. `doi:10.18653/v1/2020.acl-demos.14`.

21 Youngbin Ro, Yukyung Lee, and Pilsung Kang. Multi^2OIE: Multilingual open information extraction based on multi-head attention with BERT. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1107–1117, Online, November 2020. Association for Computational Linguistics. `doi:10.18653/v1/2020.findings-emnlp.99`.

22 Gabriel Stanovsky, Ido Dagan, et al. Open ie as an intermediate structure for semantic tasks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 303–308, 2015.

23 Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and Ido Dagan. Supervised open information extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 885–895, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. `doi:10.18653/v1/N18-1081`.

24 Aaron Steven White, Drew Reisinger, Keisuke Sakaguchi, Tim Vieira, Sheng Zhang, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. Universal decompositional semantics on Universal Dependencies. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1713–1723, Austin, Texas, November 2016. Association for Computational Linguistics. `doi:10.18653/v1/D16-1177`.

25 Mohamed Yahya, Steven Whang, Rahul Gupta, and Alon Halevy. ReNoun: Fact extraction for nominal attributes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 325–335, Doha, Qatar, October 2014. Association for Computational Linguistics. `doi:10.3115/v1/D14-1038`.

26 Alexander Yates, Michele Banko, Matthew Broadhead, Michael Cafarella, Oren Etzioni, and Stephen Soderland. TextRunner: Open information extraction on the web. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 25–26, Rochester, New York, USA, April 2007. Association for Computational Linguistics. URL: `https://www.aclweb.org/anthology/N07-4013`.

27 Junlang Zhan and Hai Zhao. Span based open information extraction. *arXiv preprint*, 2019. `arXiv:1901.10879`.

Inconsistency Detection in Job Postings

Joana Urbano[1]
skeeled, Bascharage, Luxembourg
Artificial Intelligence and Computer Science Laboratory (LIACC), Porto, Portugal

Miguel Couto
skeeled, Bascharage, Luxembourg

Gil Rocha
Faculty of Engineering, University of Porto, Portugal
Artificial Intelligence and Computer Science Laboratory (LIACC), Porto, Portugal

Henrique Lopes Cardoso
Faculty of Engineering, University of Porto, Portugal
Artificial Intelligence and Computer Science Laboratory (LIACC), Porto, Portugal

Abstract

The use of AI in recruitment is growing and there is AI software that reads jobs' descriptions in order to select the best candidates for these jobs. However, it is not uncommon for these descriptions to contain inconsistencies such as contradictions and ambiguities, which confuses job candidates and fools the AI algorithm. In this paper, we present a model based on natural language processing (NLP), machine learning (ML), and rules to detect these inconsistencies in the description of language requirements and to alert the recruiter to them, before the job posting is published. We show that the use of an hybrid model based on ML techniques and a set of domain-specific rules to extract the language details from sentences achieves high performance in the detection of inconsistencies.

2012 ACM Subject Classification Computing methodologies → Natural language processing; Applied computing → Enterprise ontologies, taxonomies and vocabularies

Keywords and phrases NLP, Ambiguities, Contradictions, Recruitment software

Digital Object Identifier 10.4230/OASIcs.LDK.2021.25

Funding *Gil Rocha*: Gil Rocha is supported by a PhD grant (with reference SFRH/BD/140125/2018) from Fundação para a Ciência e a Tecnologia (FCT).
Henrique Lopes Cardoso: This research is partially supported by LIACC (FCT/UID/CEC/0027/2020), funded by FCT.

Acknowledgements We want to thank Catarina Correia for the valuable contribution she made in the initial phase of this project.

1 Introduction

AI-based recruitment tools automate parts of the recruitment process. One of these parts is the prescreening of candidates, where a given applicant is matched against a job description using a machine learning algorithm that predicts whether or not this applicant is suited for further analysis by the recruiter.

Recruitment software often allows job information to be entered by the recruiter in textual descriptions of the job requirements and/or in structured fields that the recruiter must fill-in but that may or may not be visible to the applicant [1]. As an example, the recruiter may ask in the textual description for "good knowledge of English" and then fill-in structured fields on language requirements with "English" and language level "B2". As another example, the

[1] Corresponding author

recruiter may write "very experienced in the astrophysics domain" in the textual description and then select "10+ years of experience" in a structured field. This approach suits well recruiters, as they are very used to traditional textual job descriptions; however, they need to be able to fill-in structured fields in a way that is consistent with the textual description. The problem that arises is that there are often ambiguities and contradictions between the textual descriptions and the structured fields, which makes it challenging for AI-based algorithms, applicants and fellow recruiters to correctly interpret the job offer.

Based on the knowledge gathered by analysing the most frequent inconsistencies in a corpus of job descriptions, we developed a Natural Language Processing (NLP) model that uses both Machine Learning (ML) and rules to detect contradictions and ambiguities in job descriptions. With such a model, recruitment software is able to alert the recruiter of these inconsistencies before a job posting is published. In this paper, we present this model, focusing on the description of language requirements in job postings written in the English language. The proposed approach is currently being extended into the detection of inconsistencies in other components of job descriptions, such as the description of fields of study. The main contributions of this paper are: 1) the identification of the most common types of ambiguities and contradictions that occur when describing language requirements in job descriptions, resulting from our thorough analysis of a corpus of about 1,500 job descriptions comprising different roles, industries and clients; and 2) our NLP-based model that uses ML and a small set of rules, that has shown high performance in the detection of inconsistencies in job postings.

2 Contradictions and Ambiguities

When addressing the inconsistencies that may arise between the textual descriptions and the structured fields of a job posting, we distinguish between *contradictions* and *ambiguities*. In this section, we provide a common definition for these two types of inconsistencies and then we review how the NLP community addresses the detection of such inconsistencies.

Contradictions

We consider that a *contradiction* happens between two pieces of information when the probability of both being simultaneously true is extremely unlikely [12]. As an example applied to the description of language requirements in job descriptions, there is a contradiction when the recruiter writes that "it is required proficiency in English" and then s/he sets the English language level to A2, because these two pieces of information are not pragmatically aligned, as A2 is associated to a basic understanding of a language [13].

There are different types of contradictions being addressed by the NLP community. Marneffe, Rafferty, and Manning propose a typology of contradiction classes including *antonym*, *negation*, *numeric*, *factive*, *structural*, *lexical* and *world-knowledge* types [12].

Ambiguities

We consider that one sentence is *ambiguous* when it can have more than one possible interpretation [20], which can cause uncertainty to the reader. In this sense, the ambiguity is self-contained in a textual sentence and does not depend on the relation of this sentence with other fields, contrary to what usually happens with contradictions. An example of such an ambiguity is the sentence "You must be English/French fluent". Here, it is not clear if the candidate must be fluent in both English and French, or if it is enough to be fluent in just one of these two languages.

There are different types of ambiguities that are addressed by the NLP community, although most of them arise when a word or sequence of words have different meanings, in the same or in different contexts, with no other differences at the grammatical level (e.g., *lexical*, *pragmatic*, *semantic*), when they allow for more than one grammatical structure or different groupings of adjacent words (e.g., *syntactic*, *surface structuring*), or where the sequence of words admit borderline cases (*vagueness*) [4, 17, 20].

2.1 Related Work

The literature for detecting contradictions in text using NLP is still relatively scarce. Marneffe, Rafferty and Manning [12] present a typology of contradictions and propose an NLP-based system to automatically detect contradictions between two different sentences. Their approach converts both sentences into typed dependency graphs that are aligned in order to extract different contradiction-related features. Then, a logistic regression model is trained over these features to learn the contradiction value. Li, Qin and Liu [10] propose a convolutional neural network model to learn contradiction-specific word embedding (CWE), arguing that the use of CWE outperforms context-based word embeddings in the detection of contradictions. A different approach is provided by Pham, Nguyen and Shimazu [14], who propose a rule-based system to detect contradictions based on shallow semantic representations and binary relations extracted from sentences. Finally, Dragos [6] proposes a system to detect contradictions between two sentences that uses fuzzy semantics and that implies the estimation of the certainty of the statements, allowing to distinguish between contradictions derived from disagreement and those derived from conflictual opinions.

The use of NLP to detect ambiguities in text has some expressiveness in the Requirements Engineer domain. Gleich, Creighton, and Kof [9] use Part-Of-Speech (POS) tagging to detect the use of passive voice, adjectives and adverbs and then check for ambiguity patterns at the word level in requirements documents. Rosadini et al. [17] use POS tagging and shallow parsing to design patterns used to detect anaphoric ambiguities, vague terms, passive voice, and undefined terms in manufacturing requirements documents. They conclude that ambiguities requiring domain knowledge are hard to detect using rule-based approaches. Sabriye and Zainon [18] also use POS to detect syntactic ambiguities in software requirements documents. More concretely, they consider that a given sentence is ambiguous if it generates more than one parse tree or if it contains any "AND" or "OR" conjunctions. Rojas and Sliesarieva [16] use syntactic parsing in conjunction with regular expressions to identify vague adverbs and other ambiguous phrase structures, as well as dictionaries (WordNet [7] and VerbNet [19]) to identify ambiguous verbs. Ferrari, Donasi and Gnesi [8] study how specific words from the Computer Science lexicon vary in terms of ambiguity in different domains. For this, they built specific word embeddings from distinct vector spaces constructed in different document categories (e.g, Electronic Engineering and Medicine) and measure the variation of meaning of the CS terminology within these categories.

Most of the approaches we overview in this section use NLP to build patterns to detect contradictions and ambiguities in a way that is independent of the domain. To the best of our knowledge, our approach is the first to address the detection of ambiguities and contradictions in language requirements of job descriptions and to propose a typology of contradictions and ambiguities in language requirements.

3 Inconsistency Model Architecture

In this section, we present the general architecture of our model to detect inconsistencies in the description of language requirements in job postings written in English.

3.1 Language Requirements Specification

In this work, we consider that a job description may specify zero or more *required* languages and zero or more *optional* languages. Optional languages are only described in the textual description of the job posting, whereas required languages may be specified both in the textual description and in the structured fields of the job posting. Moreover, we consider the possibility of specifying *alternative* languages. To better understand these concepts, let us analyze the following example of a textual description.

▶ **Example 1.** The candidate must have experience in the domain and a masters in biology, biochemistry or related areas. Be very organized. We expect good knowledge of English and similar knowledge of either French or Portuguese; German is considered an asset.

With this description, the recruiter wants candidates to have good level of, at least, two languages, one being English and the other being either French or Portuguese. French and Portuguese are then alternative languages, and the minimum number of required languages is two. The recruiter also indicates that language skills in German are optional. This information must also be defined in the following structured fields, for each one of the languages mentioned in the textual description (cf. Table 1):

- *language*: the two-letter language code [2]. Sentences such as "Any other language will be considered an asset" do not require any entry in structured fields, although some recruiters may enter specific optional languages.
- *level*: the minimum language level required for the language.[2]
  In this example, we assume that good knowledge of a language corresponds to a B2 level, but our model will allow for some degree of flexibility in the definition of language levels as different recruiters may have different understandings of how to assign these levels. The specification of a language level is mandatory, even if there is no reference to such in the textual description, as it happens with the definition of German, in this example.
- *optional*: this defines whether the language is considered optional or not.
- *alternative*: this applies to non-optional languages only, and allows to distinguish between the languages that the applicant must definitely know ("no") and those that are considered alternatives ("yes").

Table 1 Example of the definition of structured fields.

language	level	optional	alternative
en	B2	no	no
fr	B2	no	yes
pt	B2	no	yes
de	B1	yes	–
Required languages: 2			

[2] Levels A1 and A2 are basic levels, B1 and B2 intermediary, and C1 and C2 proficient levels [13].

Looking again at this example, an applicant with B2 or upper levels in English and French would fit the required languages, as would an applicant with B2 or upper levels in English and Portuguese, or even these levels in English, French and Portuguese. However, an applicant with B2 or upper levels in English and Spanish or in English and German would lack one required language. In the same way, an applicant with B1 or lesser in any of the non-optional languages would be considered unfit in terms of the language requirements.

As a final note, it is frequent that recruiters put less information on the textual description than in structured fields. As an example, structured fields may specify English and French as required languages but no references to these languages are made in the textual description. As another example, the structured fields may require English, French, and German as required languages, but the textual description only mentions that "a very good understanding of German is an absolute need". These two examples show a discrepancy between textual and structured fields, but we do not consider them as being contradictory or ambiguous in this work. However, if a non-optional language is specified in the textual description and it is not specified in the structured fields, this would be a contradiction.

3.2 Language-related Inconsistencies

We analyzed a corpus of more than 1500 job postings written in English by different recruiters of different companies, concerning different roles in very distinct domains and industries (e.g., automotive, restaurants, hotels and leisure, health, air and ground transportation, biotechnology, banking, education, to name just a few). After isolating the language-related sentences, we analyzed them manually looking for ambiguities and/or contradictions within textual descriptions or between the textual descriptions and the structured fields. Backed on prior work, this allowed us to build our own terminology for the most representative types of ambiguities and contradictions in language requirement descriptions, which we present next.

- **Language-not-specified contradiction.** This occurs when a given language is mentioned in the textual job description but is not listed in the correspondent structured fields for required or optional languages.
- **Language-not-required contradiction.** This occurs when a language is identified in the required languages structured fields but is referred to as an optional language in the textual description (e.g., listing French as required and then writing "French would be an advantage" in the textual description).
- **Language-not-optional contradiction.** This is the reverse of the previous contradiction: a language is listed as optional in the structured fields while being referred to as required in the textual description (e.g., listing Italian as optional and writing "Italian is a must" in the textual description).
- **Lexical contradiction.** This occurs when the required language levels described in job textual requirements are not aligned with the language levels specified in structured fields. An example of such a contradiction is when a textual description asks for "fluency in English" and the structured fields specify B1 as the minimum level accepted for English.
- **Numerical contradiction.** This occurs when the number of minimum required languages described in the textual description does not correspond to the one specified in the structured fields. An example is when the job mentions "You are fluent in English as well as either Norwegian or Swedish" and then specifies three minimum required languages in structured fields. In fact, the textual description only refers to two minimum required languages, one mandatory and the other selected from two alternative languages.

- **Alternative-language contradiction.** This occurs when the number of alternate languages is not the same in textual and structured fields. An example is when the textual description is "You are fluent in English as well as either Norwegian or Swedish" and then in the structured fields all languages are considered as non-alternative.
- **Ambiguity.** This occurs in sentences that can have more than one possible interpretation or that are somewhat vague. For example, sentence "Good knowledge of Portuguese and preferably good knowledge of Spanish" is not clear about whether the term "preferably" applies to the language (and then Spanish would be optional) or to the desired language level (i.e., Spanish would be required and the desired level would be good). As another example, the sentence "You must have fluency in English and French or Dutch." is unclear regarding its precise meaning: on the one hand, it can be read as requiring two languages, one of them being English and the other being either French or Dutch; on the other hand, it can be interpreted as either requiring Dutch or requiring both English and French.

3.3 Model Architecture

In order to detect the ambiguities and contradictions referred to in the previous section, we developed a four-step model that combines NLP, ML and rules, as summarized in Figure 1.

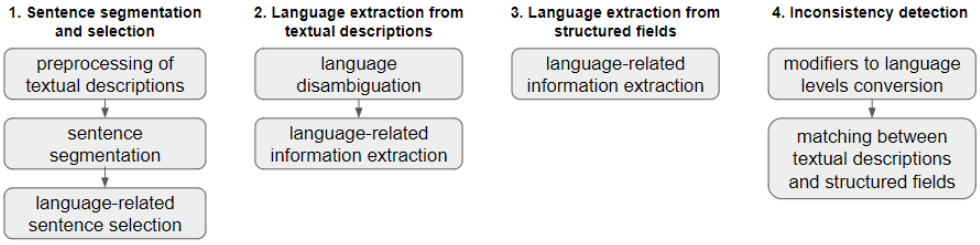


Figure 1 Overview of the Inconsistency Detection Model for language-related requirements.

The model starts by preprocessing the jobs' textual description and identifying the sentences related to language requirements, as detailed in Section 4). At step 2, it extracts the languages mentioned in these sentences, the modifiers used to describe the language levels and the remaining language attributes described in Section 3.1, including the possible existence of ambiguities (Section 5). At step 3, the model extracts language-related information from the structured fields. Finally, at step 4, the model compares the information extracted from the textual descriptions with the information from the structured fields to detect any contradictions that may exist (Section 6).

4 Sentence Segmentation and Selection

4.1 Preprocessing of Textual Descriptions

The first step of the model is the preprocessing of the textual descriptions of job postings. As these descriptions are inserted in our web-based recruitment tool, the preprocessing step consists of common NLP tasks such as the removal of URLs, symbols, HTML tags and HTML entities, and the trimming of white spaces.

4.2 Sentence Segmentation

Once cleaned, textual descriptions are segmented into sentences using a third-party NLP tool [3]. Example 2 illustrates the segmentation of the textual description of Example 1 into four distinct sentences. In this phase, some of the resulting sentences may occasionally consist of punctuation marks or single characters, which are removed by our model as they are meaningless. Also, we consider that semi-colons do not break up a sentence.

▶ **Example 2.** Result of the segmentation of the textual description of Example 1:

1. The applicant must have relevant experience in the domain and, at least, a masters in biology, biochemistry, materials science or related areas.
2. S/he must be autonomous and very organized.
3. Experience in statistics is valued.
4. We expect good knowledge of English and similar knowledge of either French or Portuguese; German is considered an asset.

4.3 Language-related Sentence Selection

After segmentation, the model identifies and selects the sentences that are related with language requirements. We tackle this task as a ML binary classification problem. For that, we trained a Random Forest model [5] using a corpus of real job sentences written in English, where each one of these sentences was labeled by us as either 1 (contains at least one mention to a language requirement) or 0 (does not contain any language requirement). The choice of the ML approach at this stage was due to the fact that it generalizes better to a more abrangent model of detection of inconsistencies where there is the need to isolate sentences related to different concepts, such as languages, education, and experiences. Also, we believe that ML may provide contextual benefits in the presence of ambiguous cases (e.g., "proficient in IT" makes it clear that the sentence refers to "Italian" and not to "Information Technology") or in extracting the best weights to apply to different types of words that are relevant to identify a language (an example of such words is "excellent", which appears frequently associated with a language but that can be used in a different context).

Each sentence of the corpus was lowercase and stemmed and, then, represented as a vector of a fixed size, corresponding to the number of features used in the model. These features are words that occur more frequently in language-related sentences, such as language names and language codes, language modifiers such as "fluent", "proficient" and "native", language-related verbs such as "speak", "write" and "read", and language related terms like "level" and "language". The resulting vectorized sentences accounted for the existence (1) or absence (0) of each one of the features in the sentences (one-hot encoding), as illustrated in Figure 2 for sentences "We expect good knowledge of English language" and "Experience in statistics is valued". The vectors shown in this figure are simplified for the sake of clarity.

	english	french	portugues	en	fr	knowledg	languag	⋮
we expect good knowledg of english languag	1	0	0	0	0	1	0	...
experienc in statist is valu	0	0	0	0	0	0	0	...

Figure 2 Simplified example of vectorized sentences.

Finally, we split the data into training and test sets and trained a Random Forest model to learn the task of classifying sentences as either containing at least one mention to a language requirement or not containing any language reference.

4.4 Validation

In this section, we evaluate our model to select language-related sentences from a corpus of textual descriptions of job postings. It does not include the evaluation of the accuracy of the sentence segmentation, as this segmentation was mostly done by a third-party tool [3].

We built up a set of 5149 sentences that resulted from the segmentation of 566 textual description of job requirements written in English, from our recruitment corpus. 574 of these sentences contained at least one mention to a language requirement. We split this dataset into train and test sets using an approximate ratio of 8:2. Table 2 summarizes the distributions of sentences, jobs and positive cases (sentences mentioning at least one language requirement) in both sets. The distribution of positive cases followed the train/test ratio.

Table 2 Train and test instances for step 1 of the model.

	sentences	**jobs**	**positive cases**
train	4267	478	490
test	882	88	84

We evaluated this model in the test set. The results have shown 99.21% of accuracy, 95.24% of recall and 95.81% in the f1-score. Because the correct identification of language-related sentences is the focus of this step of our model, we paid particular attention to the *recall* metric. We performed an error analysis on the four cases where the model failed to detect a reference to a language. Three of these cases were similar to the sentence "other languages are an asset" and have no immediate impact on the detection of inconsistencies. The fourth case is "Luxembourgish is an asset", and it is due to the case that "Luxembourgish" appears frequently in our corpus associated to other concepts than languages (e.g., "Luxembourgish law", "Luxembourgish offices"). We must address this issue in future work.

5 Language Extraction from Textual Descriptions

The second step of our model applies to the sentences identified as mentioning at least one language requirement in the previous step. For each one of these sentences, the model extracts the language names appearing in the sentence, their corresponding language-level modifiers, should they exist (e.g., "fluent", "native"), whether the languages are optional or non-optional, and whether non-optional languages are considered alternative or not. The model also derives the number of required languages stated in each sentence and whether the sentence is ambiguous or not. An illustrative example is given in Table 3.

Table 3 Example of information extracted in Step 2 for sentence "Excellent English skills, both written and verbal, and a fluent knowledge of French and Dutch".

language code	**modifier**	**optional**	**alternative**
en	excellent	no	no
fr	fluent	no	no
nl	fluent	no	no
Required languages: 3, Ambiguity: no			

5.1 Language disambiguation

The extraction algorithm starts with a *language disambiguation phase*, where sentences are searched for specific terms that could denote either a language code or some other unrelated concept. As an example, the term "HR" is commonly used in the recruitment domain as an acronym for "human resources" but it can also be the two-letters code for the Croatian language. Similarly, the term "IT" can be an acronym for "information technology" or the language code for Italian, and the term "PL" can be the acronym for "programming language" or it can be the language code for Polish. In order to cope with these situations, we built a controlled vocabulary of words related to these possible other meanings of specific language codes. For instance, to cope with the "PL" term, we added to this vocabulary words related with programming languages, such as "R", "Python", "Java", and "C". Then, in the presence of one such ambiguous language code in the sentence under analysis, the algorithm searches for one or more occurrences of the related terms in the vocabulary, and if occurrences are found the algorithm drops the term from the list of possible languages.

5.2 Modifier detection

After the extraction of languages, the algorithm proceeds with the detection of the language level modifiers associated with each one of the languages identified in a sentence.

We verified that the standalone use of dependency parsing or part-of-speech techniques to automatically extract these language level modifiers led to different types of errors, mostly when there are different languages and modifiers in the same sentence or when sentences are verbless (e.g., "Fluent in French and English proficiency"), which are common in the description of language requirements. On the other hand, a thorough analysis of the positioning of modifiers in hundreds of language-related sentences of real job postings in our corpus allowed us to identify different patterns of distancing and positioning between languages and modifiers that could be converted into syntactic rules to extract these language modifiers, or level indicators. As an example, Figure 3 illustrates the positioning of the modifier "excellent" relative to language "English" and the positioning of modifier "fluent" that affects both languages "French" and "Dutch".

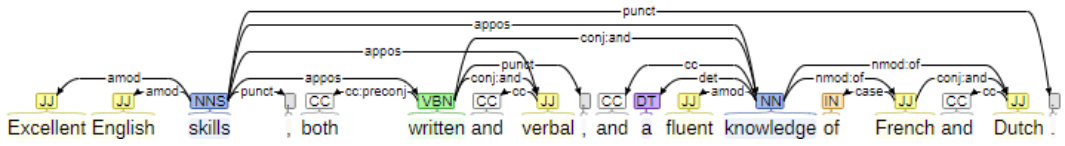


Figure 3 Use of dependency parsing to measure the distances between languages and modifiers in sentence "Excellent English skills, both written and verbal, and a fluent knowledge of French and Dutch". [3]

Next, we present some of the patterns of modifiers' identification, positioning and distancing to the correspondent languages that were more common in our corpus and that dictated our rule-based model.

1. *The number of language modifiers is relatively small.* We verified that the modifiers used by different recruiters in different roles and domains do not vary substantially. Therefore, we built a list of possible modifiers that the algorithm should look at, starting with a vocabulary of modifiers that we found in our recruitment corpus (e.g., "fluent",

[3] Figure generated using CoreNLP tool [11].

"proficient", "solid") and manually extending it with relevant synonyms with the help of NLP tools such as WordNet [7] and qdap [15]. This list was later validated by our recruitment experts.

2. *Some modifiers comes immediately before the language*, as in "We require good English".
3. *Some modifiers come before the language but not immediately.* An example of this pattern is "Fluent in writing and reading in Norwegian". In this case, we verified that the words between the modifier and the language could be removed without impairing the correct identification of the modifier, resulting in a sentence matching Pattern 2. We created a list with such *neutral* words (e.g., "writing", "reading", "skills" and "presentation"), augmented with synonyms of these obtained from WordNet and qdap.
4. *Some modifiers come immediately after the language*, as in "Speak English fluently".
5. *Some languages have more than one modifier.* An example of this is "Good knowledge of English", where both "good" and "knowledge" are on the list of possible modifiers. We identified specific words, such as "knowledge", "communication", and "skills", which are ignored by our extraction model when they appear adjacent to other modifiers.
6. *Some modifiers apply to more than one language separated by connective "and".* An example of this pattern is "Workable English and French".
7. *Some languages do not have a modifier*, as in "Swedish is mandatory".
8. *Some sentences have duplicated languages but just one possible modifier.* One example is "Native level of spoken and written English and Spanish workable knowledge (English is of utmost importance)". In this sentence, "English" is duplicated, but neither one of the terms in "is of utmost importance" appear in our list of possible modifiers. Therefore, the modifier associated to English extracted by our model is "native".

5.3 Detection of optional, alternative and required languages

After language and modifier extraction, the algorithm checks which languages are optional/ non-optional, and which non-optional languages are alternatives. We found the following patterns:

1. *The description of optional languages is often accompanied by specific terms*, as in "French and English; Spanish and Portuguese being an advantage", where the term "advantage" makes obvious that Spanish and Portuguese are optional languages. In reality, the analysis of our corpus has shown that a simple search for specific words (e.g., "asset", "advantage", "plus") proved to be an efficient approach, so we built a list with such *optional* words extended by synonyms.
2. *The description of non-optional languages is often accompanied by specific terms*, such as "mandatory", "required", "must", "compulsory" and "essential", as in "Swedish is mandatory". We built a list of such *non-optional* words, extended with synonyms, and consider a language as non-optional if it is associated to any term of this list.
3. *Alternative languages are often associated with specific terminology (e.g., "either", "as well as", "combined with", and "together with"), connectives ("or") and related indicators ("/").* Examples of these are "You should master French or English" and "English and either French or German". We use this expressions to extract alternative languages.

5.4 Ambiguity and number of required languages detection

Some language-related sentences are ambiguous and the extraction algorithm should not only signal these ambiguities but also take decisions in the presence of these ambiguities. As an example, the sentence "Profound written knowledge of English, Dutch and/or German

are required, Luxembourgish is a plus" raises a certain level of ambiguity concerning the *minimum* number of required languages, which could be either one (English or Dutch or German) or two (English and either Dutch or German). As a different example, the sentence "You must be English/French fluent" raises an ambiguity concerning the meaning of "/" in this sentence, which could indicate that the recruiter wants the applicant to know English *or* French or English *and* French. However, if we consider the sentence "You must be English/French bilingual", the term "bilingual" helps to disambiguate the sentence. As a final example, the algorithm understands the sentence "Fluency in English and Spanish or French" as corresponding to two minimum required languages, all non-optional, and two alternative languages (Spanish and French), but it also raises an ambiguity as this sentence allows for different interpretations. When all information about the optional/non-optional and alternative languages is extracted, the number of *minimum* required languages described in a sentence is computed as the sum of all non-optional languages, to which a unit is added if in the presence of, at least, one alternative language.

5.5 Validation

To evaluate the extraction algorithm, we annotated a set of entries corresponding to 733 language requirements (715 of these specifying a specific language) in 371 language-related sentences, from 302 textual description of job postings. More concretely, for each language mentioned in each sentence (an *entry*), we annotated the language code, its modifier, whether the language was optional or not and whether the language was alternative or not. At the sentence level, we annotated the number of required languages mentioned in the sentence and whether or not the sentence was ambiguous. Table 4 further characterizes this dataset, presenting the frequency of the labels for each one of these components. As a note, "–" values are related to sentences that mention a language requirement without specifying a language name, and "others (n)" aggregate the frequencies of n infrequent values.

Table 4 Frequency table for the labels used in step 2.

modifier	fluent: 31%, (no modifier): 24%, fluency: 10%, excellent: 9%, participate: 4%, good: 3%, –: 2%, basic: 1%, communication: 1%, knowledge: 1%, very good: 1%, outstanding: 1%, proficiency: 1%, strong: 1%, others (22): 10%
language	EN: 42%, FR: 29%, DE: 10%, LB: 4%, NL: 3%, –: 2%, others (18): 10%
optional	no: 82%, yes: 15%, –: 2%
alternative	no: 89%, yes: 11%
required	2: 63%, 1: 30%, 3: 5%, 4: 1%, 0: 1%
ambiguity	no: 94%, yes: 6%

We split this dataset into training and test sets using a 7:3 ratio, where the training data was used to develop the rules and the test set to validate the model. The number of entries, sentences and jobs for the training and test sets is shown in Table 5. We run our rule-based extraction algorithm on the test set, for each one of the extraction phases. Errors in one phase do not propagate into the following one in this test settlement, except for error in the extraction of optional languages, which propagates into the extraction of the *minimum* required languages of the job posting and into the extraction of ambiguities.

We obtained 100% of accuracy in the extraction of the language names. Table 6 presents the results obtained for the extraction of the optional, alternative and ambiguity features, and Table 7 presents the errors per class and the accuracy for the extraction of modifiers

Table 5 Train and test instances for step 2.

	entries	sentences	jobs
train	529	262	216
test	204	109	86

and for the *minimum* number of required languages per sentence.

Table 6 Results for step 2 for binary classes.

	accuracy	recall	f1-score
Optional/Non-Optional	94.9%	98.04%	96.77%
Alternative/Non-Alternative	98.09%	100%	98.96%
Ambiguity	94.18%	83.33%	66.67%

Table 7 Accuracy and number of errors for modifiers and number of required languages.

label	#errors	accuracy
modifier	fluent: 4, fluency: 4, good: 1, knowledge: 1, others: 0	94.90%
required languages	0: –, 1: 0, 2: 1, 3: 1, 4: 0	97.67%

From the results, we associate the perfect accuracy of the model in extracting the language names to the fact that recruiters tend to write languages and language codes correctly, however, we intend to reinforce typos checking in a future version of the model.

An important result to look for is the ability of the extraction model to distinguish between optional and non-optional languages, because errors on this phase propagate to posterior phases of the model (in a production environment), such as the detection of alternative languages and the computation of the *minimum* number of required languages per sentence. Therefore, we were looking for high values of recall, and our model was able to detect the non-optional languages with a value of recall of 98.04%. An error analysis on unsuccessful cases have shown that some of them are hard to detect because of structural malformations of the sentence – e.g., "Portuguese and Spanish (of advantage)". In some other cases, we verified that an additional rule will be needed to be added to our set of rules. Finally, we observed that the accuracy of the *minimum* number of required languages per job posting was 91.86%, reflecting the propagation of errors from the optional/non-optional phase.

The results have also shown high performance in the extraction of alternative languages, with 98.09% of accuracy and 100% of recall. Concerning the extraction of modifiers, our model achieved an accuracy of 94.90%. The error analysis have shown that the errors were mainly associated to the existence of multiple modifiers assigned to the same language and to the existence of large distances between languages and modifiers. We intend to address the tuning of our neutral-words skipping process, as well as the possibility to add another rule to fix these errors, in future work. Finally, the algorithm for the extraction of ambiguities achieved an accuracy of 94.18% and 83.33% of recall, with one single false negative in sentence "Portuguese and Spanish (of advantage)". The majority of false positives propagated from the detection of optional languages phase. It also become evident to us that we have to add more ambiguous sentences to our dataset in a future evaluation of our inconsistency model.

6 Inconsistency Detection

The third step of our inconsistency detector model is the detection of contradictions in language requirements, by comparing the information extracted from textual descriptions to the information specified in the structured fields, and by assigning a specific type of inconsistency (cf. Section 3.2) to the inconsistent sentences.

6.1 Conversion of modifiers to language levels

This step starts with the conversion of the language modifiers extracted by the model into language levels, as it is the way this information is present in structured fields. As an example, "native [English]" must be converted into level C2, and "basic knowledge [of French]" must be converted into A2. In order to make these conversions, we asked our recruitment experts to categorize all modifiers of our extended modifiers list into categories of similar semantics, using the technique of card sorting [21]. At the end of this process, our experts assigned a language level (from A1 to C2) to each group of similar modifiers, and indicated the *minimum* and *maximum* levels that could be associated to a given language level without being considered *contradictory* to that language level. As an example, the modifiers "fluent", "proficient", "perfect", and "flawless" were assigned a *minimum* level of C1 and a *maximum* level of C2. Therefore, if a given sentence mentions "Perfect Spanish"and the associated level in structured fields is B2, this would raise a *lexical* contradiction. As card sorting was done separately for each one of our experts, at the end of this process they met to discuss the modifiers that were grouped differently or for which the level assignment was different and a decision was made by consensus. The result of this process is a validated dictionary of modifiers-language levels correspondences that our model uses in this initial stage.

6.2 Matching between textual and structured fields

The model proceeds by matching each item of the extracted information with the corresponding item of the structured fields, using a set of rules that allow the model to raise zero or more contradictions. Next, we present a set of examples illustrating the model's rules.

▶ **Example 3. The applicant must be fluent in English and in French**

Structured Fields				Textual Description			
language	**level**	**optional**	**altern.**	**language**	**modifier**	**optional**	**altern.**
en	B2	no	no	en	fluent	no	no
				fr	fluent	no	no
Required languages: 1				Required languages: 2, Ambiguity: no			

Here, the textual description refers to a language that is not defined in the structured fields (French), which also implies that the number of required languages does not match, and the model raises the *Language-not-specified* and *Numerical* contradictions.

▶ **Example 4. Fluent in Norwegian and/or Italian and English**

Structured Fields				Textual Description			
language	**level**	**optional**	**altern.**	**language**	**modifier**	**optional**	**altern.**
no	B2	no	yes	no	fluent	no	yes
it	A2	no	yes	it	fluent	no	yes
en	B2	no	no	en	fluent	no	no
Required languages: 2				Required languages: 2, Ambiguity: yes			

This is one typical example that raises an *Ambiguity* alert. In this case, the model assumes that there are at least two non-optional languages, one being English and the other being either Norwegian or Italian, and we verify that the logic associated with the precedence of the connectives ("and", "or") and with the use of "/" is accompanied by the *world knowledge* that English is an Universal Language. This would not the case in other situations, so the use world knowledge in inconsistencies detection should be addressed in future work. Finally, the model would raise a *Lexical Contradiction* because the textual description mentions that the applicant must be fluent in one of the alternative required languages (Norwegian or Italian) and the language level associated to Italian is A2, which corresponds to a basic level, below the *minimum* language level associated with modifier "fluent" by our recruitment experts.

▶ **Example 5. English is mandatory**

Structured Fields				Textual Description			
language	**level**	**optional**	**altern.**	**language**	**modifier**	**optional**	**altern.**
en	A1	no	no	en		no	no
Required languages: 1				Required languages: 1, Ambiguity: no			

In this example, the model does not raise any contradiction or ambiguity. As the model does not extract any modifier for English, it would not check for lexical contradictions.

▶ **Example 6. Work knowledge of German or Dutch**

Structured Fields				Textual Description			
language	**level**	**optional**	**altern.**	**language**	**modifier**	**optional**	**altern.**
de	B2	no	no	de	knowledge	no	yes
nl	B2	no	no	nl	knowledge	no	yes
Required languages: 2				Required languages: 1, Ambiguity: no			

This is another typical example appearing in our corpus. The textual description asks for one required language but the structured fields defines two required languages, ignoring that both German and Dutch should be defined as alternative languages. In this case, our model raises an *Alternative-languages contradiction* and a *Numerical contradiction*.

6.3 Validation

We validated this step of our model with a dataset composed of 353 language-related sentences from 302 textual descriptions of job postings, corresponding to 715 instances of languages (*entries*). Table 8 presents the frequency table for each type of contradiction. The number of entries, sentences and jobs for the training and test sets is shown in Table 9.

Table 8 Frequency of each type of contradiction in the dataset of step 4.

Lang-not-spec	Lang-not-req	Lang-not-opt	Lexical	Mandatory	Numeric
5%	3%	2%	10%	3%	15%

We run our algorithm in the test set and it achieved an accuracy of 100% in detecting contradictions of types *Language-not-specified*, *Language-not-required*, *Language-not-optional*, *Alternative-language* and *Numerical* (per posting). Regarding the detection of *Lexical* contradictions, the model was accurate 98.98% of the times, reaching a recall of 91.30% and a f1-score of 95.45%. From error analysis, we verified that the errors in this phase were related to cases where the recruiters were more exigent in the language requirements as specified

Table 9 Train and test instances for step 3 of the model.

	entries	sentences	jobs
train	519	252	216
test	196	101	86

in structured fields than in the textual descriptions of job postings, as when they asked for "Good knowledge of English"and then asked for C1 levels in structured fields. Our team of experts concluded that this type of contradiction is not as severe as the one occurring in the opposite direction (e.g., asking for "Native English"and then defining B2 levels).

7 Conclusions

This paper presented an approach to detect contradictions and ambiguities in the description of language requirements in job descriptions written in English. We focused the content of the paper in two essential components. First, we provided and analyzed a set of examples of common language-related sentences containing at least one ambiguity or that are contradictory when compared to the language requirements specified in the job's structured fields. Then, we proposed a terminology for the description of inconsistencies in language requirements, composed of six types of inconsistencies and one ambiguity. This proposal resulted from the thorough analysis of our corpus of job descriptions from hundreds of distinct job roles published by different recruiters from several organizations, in different countries and industries.

Second, we proposed a four-step NLP-based model to detect these inconsistencies from job descriptions. This model uses machine learning to extract the language-related sentences (step 1) and a set of comprehensive rules to extract relevant information from these sentences (step 2) and to detect the inconsistencies (step 4). We have shown that even with a restricted set of rules the model achieved high performance in each one of the steps. Moreover, this model will serve as a baseline to further improvements, which can include the use of a machine learning approach to extract the mentioning languages and their requirements from sentences.

As future work, we intend to tune our existing rules to fix error cases detected in error analysis and to provide a more sophisticated approach to detect alternative languages. We also intend to enrich the NLP preprocessing with typo checking. Although we are convicted that our annotated dataset of job postings covers the majority of possible sentences describing language requirements, we still intend to extend it (namely, re-enforcing the number of ambiguous sentences), in order to evaluate how the model would scale to different types of sentences. Finally, we believe that this approach adapts well to sentences written in other languages, such as French and Portuguese, so we intend to adapt it to these languages.

References

1 Edward Tristram Albert. Ai in talent acquisition: a review of ai-applications used in recruitment and selection. *Strategic HR Review*, 2019.

2 Harald Alvestrand. Tags for the identification of languages. *RFC 1766, March*, 1995.

3 Kenneth Benoit, Kohei Watanabe, Haiyan Wang, Paul Nulty, Adam Obeng, Stefan Müller, and Akitaka Matsuo. quanteda: An r package for the quantitative analysis of textual data. *Journal of Open Source Software*, 3(30):774, 2018.

4 Daniel M Berry, Erik Kamsties, and Michael M Krieger. From contract drafting to software specification: Linguistic sources of ambiguity. *A Handbook*, 2003.

5 Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
6 Valentina Dragos. Detection of contradictions by relation matching and uncertainty assessment. *Procedia Computer Science*, 112:71–80, 2017.
7 Christiane Fellbaum. Wordnet. *The encyclopedia of applied linguistics*, 2012.
8 Alessio Ferrari, Beatrice Donati, and Stefania Gnesi. Detecting domain-specific ambiguities: an nlp approach based on wikipedia crawling and word embeddings. In *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)*, pages 393–399. IEEE, 2017.
9 Benedikt Gleich, Oliver Creighton, and Leonid Kof. Ambiguity detection: Towards a tool explaining ambiguity sources. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*, pages 218–232. Springer, 2010.
10 Luyang Li, Bing Qin, and Ting Liu. Contradiction detection with contradiction-specific word embedding. *Algorithms*, 10(2):59, 2017.
11 Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60, 2014.
12 Marie-Catherine Marneffe, Anna N Rafferty, and Christopher D Manning. Finding contradictions in text. *Proceedings of ACL-08: HLT*, pages 1039–1047, 2008.
13 Council of Europe. Council for Cultural Co-operation. Education Committee. Modern Languages Division. *Common European Framework of Reference for Languages: learning, teaching, assessment*. Cambridge University Press, 2001.
14 Minh Quang Nhat Pham, Minh Le Nguyen, and Akira Shimazu. Using shallow semantic parsing and relation extraction for finding contradiction in text. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1017–1021, 2013.
15 Tyler W Rinker. qdap: Quantitative discourse analysis package. *University at Buffalo/SUNY*, 2013.
16 Allan Berrocal Rojas and Gabriela Barrantes Sliesarieva. Automated detection of language issues affecting accuracy, ambiguity and verifiability in software requirements written in natural language. In *Proceedings of the NAACL HLT 2010 Young Investigators Workshop on Computational Approaches to Languages of the Americas*, pages 100–108. Association for Computational Linguistics, 2010.
17 Benedetta Rosadini, Alessio Ferrari, Gloria Gori, Alessandro Fantechi, Stefania Gnesi, Iacopo Trotta, and Stefano Bacherini. Using nlp to detect requirements defects: an industrial experience in the railway domain. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*, pages 344–360. Springer, 2017.
18 Ali Olow Jim'ale Sabriye and Wan Mohd Nazmee Wan Zainon. A framework for detecting ambiguity in software requirement specification. In *Information Technology (ICIT), 2017 8th International Conference on*, pages 209–213. IEEE, 2017.
19 Karin Kipper Schuler. Verbnet: A broad-coverage, comprehensive verb lexicon. *Ph. D. Thesis, University of Pennsylvania*, 2005.
20 Adam Sennet. Ambiguity. *Stanford Encyclopedia of Philosophy*, 2011.
21 Donna Spencer and Todd Warfel. Card sorting: a definitive guide. *Boxes and arrows*, 2:1–23, 2004.

A Workbench for Corpus Linguistic Discourse Analysis

Julia Krasselt
Zurich University of Applied Sciences, Switzerland

Matthias Fluor
Zurich University of Applied Sciences, Switzerland

Klaus Rothenhäusler
Zurich University of Applied Sciences, Switzerland

Philipp Dreesen
Zurich University of Applied Sciences, Switzerland

Abstract

In this paper, we introduce the *Swiss-AL workbench*, an online tool for corpus linguistic discourse analysis. The workbench enables the analysis of Swiss-AL, a multilingual Swiss web corpus with sources from media, politics, industry, science, and civil society. The workbench differs from other corpus analysis tools in three characteristics: (1) easy access and tidy interface, (2) focus on visualizations, and (3) wide range of analysis options, ranging from classic corpus linguistic analysis (e.g., collocation analysis) to more recent NLP approaches (topic modeling and word embeddings). It is designed for researchers of various disciplines, practitioners, and students.

2012 ACM Subject Classification Computing methodologies → Language resources; Computing methodologies → Discourse, dialogue and pragmatics

Keywords and phrases corpus analysis software, discourse analysis, data visualization

Digital Object Identifier 10.4230/OASIcs.LDK.2021.26

Supplementary Material *InteractiveResource (Online Tool)*: `https://swiss-al.linguistik.zhaw.ch/shiny/dashboard/`

Funding This work was supported by internal funding of the Zurich University of Applied Sciences.

1 Introduction

Linguistic corpora are highly dependent on tools that enable a systematic analysis of primary data, annotations and metadata. Corpora are always approached with the need for specific information, e.g., regarding the frequency of a word form over time or a word's embeddedness in a linguistic context. This dependency relation is reinforced by the variety of research fields that use corpora, such as discourse analysis, lexicography, or language acquisition. An in-depth technical and statistical knowledge of processing annotated language data (e.g., by means of a programming language such as Python or R) is not necessarily part of the core competencies of these research fields. The same holds true for the translation of quantitatively obtained corpus data into diagrammatic representations (e.g., bar and line graphs or networks). Thus, researchers working with corpus data need to rely on appropriate analysis tools.

Furthermore, corpora are not only an invaluable resource in linguistic research, but also in other academic disciplines and in the field of professional communication. From an applied perspective, a good and easy to understand corpus analysis tool is needed because corpus data is approached from an "outsiders" (non-linguistic) perspective, e.g., by professionals developing a communication strategy for a company.

Here, we present the *Swiss-AL workbench*, an online tool for analysing Swiss-AL, a multilingual web corpus for Applied Linguistics [17]. The motivation for the development of the workbench arose from the need for a user-friendly, intuitively accessible, state-of-the-art analysis tool for different user groups. As a consequence, the workbench is characterized by the following key aspects: (1) it is easily accessible with any online browser without prior registration; (2) it has a strong focus on visualizing results in a diagrammatic way; (3) it offers not only traditional corpus linguistic methods (e.g., collocation analysis), but also more recent approaches from natural language processing (topic modeling and word embeddings).

The Swiss-AL workbench is designed for the purpose of applied discourse linguistics, but it can also be used in other research fields. Applied discourse analysis is concerned with identifying the communicative conditions that shape the way a society talks and writes about specific topics [28]. These conditions appear as patterns of language use, i.e. recurring ways of talking or writing about something [7]. As an applied discipline, discourse linguistics pursues the goal of solving communicative problems. The Swiss-AL workbench enables both the corpus-based and corpus-driven identification of patterns of language use by providing different means to analyse the available corpora (e.g., the statistical co-occurrence of words or the distribution of ngrams).

The workbench is designed for a rather heterogeneous audience in order to overcome the difficulties and desiderata outlined in the first two paragraphs. The intended audience ranges from discourse and corpus linguists (who typically have very specific questions, e.g., regarding variants of a specific word or regarding the frequency distribution of a word over a certain period of time) to researchers from other disciplines (who do not normally have linguistic expertise), students and actors of professional communication.

The paper is structured as follows: Section 2 gives a brief overview of related work on corpus analysis software; Section 3 describes the intended audience of the workbench and typical use case scenarios. Section 4 describes the workbench, its architecture and underlying data and individual functionalities in detail. Section 5 contains a conclusion and plans for future work.

2 Related Work

The first digital tools for analysing corpora date back until the 1970s and have since then developed from merely providing concordances for a given search word to web-based or standalone software allowing for quantitative and qualitative analysis of ever-growing corpora (for a historical overview, see [21]).

Modern corpus analysis software can be categorized in (1) ready-made corpus analysis tools, i.e. tools already equipped with corpora and a set of functionalities to analyse these corpora, (2) corpus analysis software designed for the import of own corpora and (3) software allowing for both approaches. Table 1 gives an overview over existing corpus analysis tools and a comparision with the Swiss-AL workbench.

Regarding (1) and with a focus on German, the Institut für Deutsche Sprache and the Berlin-Brandenburgische Akademie der Wissenschaften provide online tools to the reference corpora DeReKo and DWDS [3, 18, 10]. Similar to the Swiss-AL workbench, these tools provide only limited access to the full texts in the corpus due to copyright reasons. [29] introduce the cOWIDplus Viewer, allowing to analyze the vocabulary of German online media during the COVID-19 pandemic on a regularly updated data base. Similar to the Swiss-AL workbench, the cOWIDplus Viewer has a focus on visualizing results and is aimed at non-linguistic experts. For English, the BYU corpus analysis tools offer access to a broad variety of corpora.

Table 1 Comparison of corpus analysis tools, with a focus on available methods and intended audience (some of the tools offer additional methods, not all can be mentioned here for pragmatical reasons).

		concordance	keywords	collocations	tm[5]	we[6]	ngrams	frequency lists	distribution analysis	text view	intended audience[1]
pre-installed corpora	DWDS[8] [3]	✓		✓				✓[3]	✓		scientific and non-scientific audience
	Cosmas II[8] [18]	✓		✓				✓	✓		researchers, translators, students, linguistic laypeople
	english-corpora.org[8]	✓	✓	✓			✓	✓	✓	✓	researchers, students
	Swiss-AL workbench[8]	✓	✓	✓[2]	✓	✓	✓	✓	✓	(✓)[4]	researchers, students, practitioners, linguistic laypeople
import of own corpora	AntConc[7] [1]	✓	✓	✓			✓	✓		✓	students
	CorpusExplorer[7] [25]	✓	✓	✓			✓	✓	✓	✓	corpus linguists, data mining experts
	CQPweb[8] [15]	✓	✓	✓				✓	✓	✓	non-technical users
	WMatrix[7] [24]	✓	✓	✓			✓	✓		✓	academic researchers and students
	Wordsmith[7] [26]	✓	✓	✓			✓	✓	✓	✓	lexicographers, researchers, students
pre-installed corpora/import of own corpora	LancsBox[7] [6]	✓	✓	✓[2]			✓	✓		✓[3]	anyone interested in language
	Sketch Engine[8] [16]	✓	✓	✓			✓	✓	✓	✓	linguists, lexicographers, translators, students, teachers

1) According to self-description in publications/on website
2) including collocation networks
3) only for specific corpora
4) on request (due to copyright restrictions)
5) topic models
6) word embeddings
7) desktop application (installed locally)
8) server based application

With regard to (2), the *Corpus Workbench* (CWB) and its webserver based GUI *CQPweb* is one of the most flexible tools for indexing and analysing own corpora [14, 15]. The Swiss-AL workbench heavily relies on the CWB architecture (see Section 4.1). Other freely available corpus tools are *AntConc* [1], *Wordsmith* [26] and *CorpusExplorer* [25].

One of the leading corpus tools that enables the import of own corpora but that is also equipped with a large variety of corpora in multiple languages is *Sketch Engine* [16]. It is a proprietary software and became a standard tool especially in lexicography. Another proprietary corpus analysis software is *WMatrix* [24], accessible online via Lancaster University. Recently, *LancsBox* has been published by the University of Lancaster as a standalone software package [6]. It is designed for importing own corpora but is also equipped with a range of preinstalled corpora. Similar to the Swiss-AL workbench, it also has a strong focus on visualizing results.

The Swiss-AL workbench presented here belongs to the first group of software tools, since it enables access to a variety of corpora from the Swiss-AL corpus family. While a wide range of tools for corpus linguistic analysis exists, the Swiss-AL workbench fills a noticeable gap: it is easily accessible (without a user account or a prior installation of software), targets at a very heterogeneous audience and offers a wide range of analysis methods.

3 Intended Audience and Use Case Scenarios

The intended audience includes three main groups. (1) Corpus linguistic laypeople use the workbench especially as project partners in discourse-related research. Typically, as practitioners of professional communication, they have very specific questions, e.g., regarding the frequency distribution of a word referring to their organization. Often for the first time, the workbench provides these practitioners with access to data from the discourse that affects them and enables a much wider perspective, i.e. extrospection [11]. It is planned to offer the workbench also for actors from the civil society like NGOs and citizen science initiatives. The workbench enables practitioners to change their perspective from introspection to extrospection.

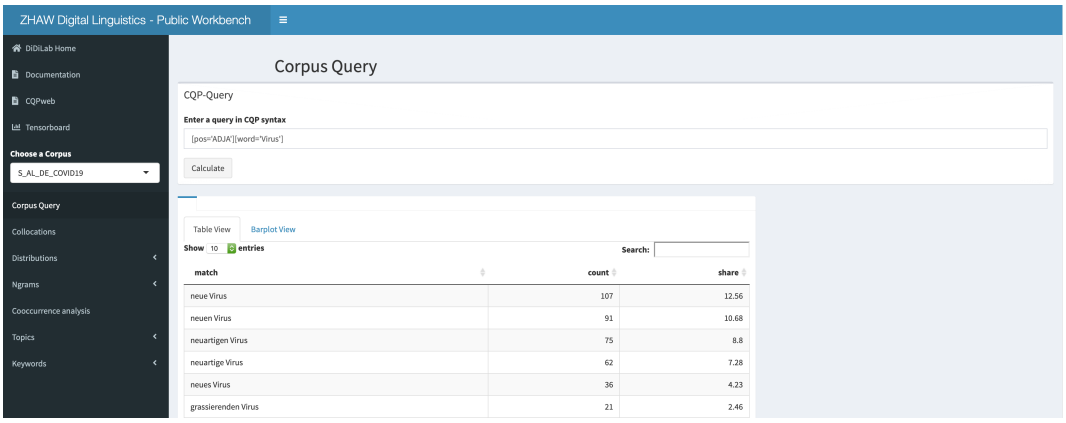


Figure 1 Workbench interface.

(2) After an introduction, undergraduate to PhD students of Applied Linguistics can use the workbench to learn the variety of corpus linguistics analysis almost independently. As such, the workbench can be used for exercises, seminar papers, bachelor, and master theses. (3) The group of corpus data experts uses the workbench especially in inter- and transdisciplinary research projects. The workbench makes it possible to quickly explain and show how corpus linguistic analysis works and how results can be visualized.

The main advantage of the workbench is the macro perspective (distant reading) on discourses, including visualizations. The workbench offers the possibility to aggregate discourses in form of distributions, e.g., of frequency and co-occurrence data (see Section 4.3). As the intended audience of the workbench is so heterogeneous, the workbench has a tidy surface (cf. Figure 1) and can be used without prior registration. The available functions are non-nested. Instead, they are available from the main interface in order for users to always be well oriented.

A typical use case scenario could proceed in three steps (of course, depending on the competencies of the user group). For example, if a user wants to analyze the communication about pandemic measures in the German and Italian COVID-19 discourse in Switzerland, a first step would be to use the word embedding model to find semantically similar words referring to pandemic measures. Alternatively, topic modeling can be used for a first overview to get hints on discoursive thematicity of known measures. As a second step, frequency and distribution over time can be analyzed for the words identified in the word embedding model. Finally, the most interesting/frequent words can be used for collocation, co-occurrence, and ngram analysis.

4 Workbench Description

The workbench is available under the following URL: `https://swiss-al.linguistik.zhaw.ch/shiny/dashboard/`. Figure 1 shows the general layout: on the left navigation pane, users can choose a corpus from a drop-down menu. The workbench provides various functions which will be performed for a selected corpus. The results will be displayed in the right window pane. For each function, different visualization options are available, e.g., a tabular view or a graph view. Additionally, the workbench is equipped with a documentation giving an overview over the available corpora and implemented corpus linguistic functions.

4.1 Workbench Architecture

The main workbench is built on top of R Shiny by RStudio [9]. R Shiny allows to create a visually pleasing web app which triggers R code on the fly and allows for adjustment and manipulation of parameters. This principle of separating the code from the visuals allows us to create a workbench that is easy to use for laypersons and linguists alike. The visualisations and queries are done in real time. The majority of the corpus-related functions are processed with the help of the polmineR package [4] which uses the underlying Corpus Workbench (CWB, [14]) for accessing the corpus data. For a more detailed description of the implemented functionalities, please see Section 4.3.

At its core, the so called shiny app triggers R functions, which in turn retrieve and process data and send them back to be rendered on the website. The data can be manipulated in various ways in order to create a useful plot or table for further investigation by the app users. In terms of manipulation, shiny can be used to give the user a choice of parameters to take into account. E.g., it is possible to offer the user a simple slider to limit the year in which the texts in a corpus were created. This allows for a better investigation and data exploration. Further, due to the usage of the polmineR package, the power of the CWB syntax can be used to create a detailed analysis of the underlying data.

All corpora available on the workbench belong to the family of corpora subsumed under the label *Swiss-AL* ([17], compare Section 4.2). The texts in these corpora are crawled from a predefined set of web pages and annotated linguistically by an automated pipeline. Since Swiss-AL mainly contains texts that are subject to Swiss copyright restrictions, the workbench currently does not offer access to the full texts in the corpora.

Since the workbench is currently in an early stage of development and due to the copyright restrictions of the underlying corpus data, the code is not open source.

4.2 Available Corpora

The workbench is equipped with a variety of corpora from the Swiss-AL family of corpora [17]. The corpora are web-based, i.e. texts are crawled from a curated list of websites from politics, media, industry, science and civil society. All corpora are processed with a linguistic pipeline (described in detail in [17]). Due to the multilingualism in Switzerland, most corpora are available in German, French, and Italian. The workbench also serves as a tool to make research data publicly available in order to follow an open research data policy. E.g., we recently published a corpus on Swiss COVID-19 discourses.

4.3 Functionalities

The workbench provides access to standard linguistic methods for discourse analysis (corpus query, distribution analysis, collocations/co-occurrences, keywords, ngrams, cf. [2, 7]) and also to approaches that have become relevant for the analysis of public discourses more recently, coming from natural language processing (topic modeling, word embeddings).

Corpus Query

This mode of analysis allows to query for a word or a sequence of words in a selected corpus by using CQP-syntax [14] and to get the frequency for this query. Strings can be specific (combinations of) word forms, lemmas, part-of-speech or even dependency relations, depending on the token level annotations available in the selected corpus (so called positional attributes). CQP-syntax allows for the use of regular expressions. To that end, users can

search for strings matching a specific pattern (see example 1). Frequencies will be reported for all matches of a query (e.g., the second search string in the example below will report all individual frequencies for words beginning with the morpheme {Vir-}).

Example 1 corpus queries using CQP-syntax.

```
[pos = "ADJA"][lemma = "Virus"] # sequence of adjective plus 'Virus'
[word = "Vir.*"] # word forms beginning with {Vir-}
[depRel = "SB" & pos = "NN"] # nouns in subject position
```

Distribution Analysis

By entering up to five word forms, users can analyse the relative frequency of these words in a user-defined time period and/or a user-defined set of sources. For example, users can get the frequencies per month for the word forms *Lockdown* and *Shutdown* in Swiss media since January 2020 in order to see wether there is a preference for one of these words and wether these preferences change over time. Results will be visualized as a line graph or barplot.

ngrams

Since a considerable amount of language consists of conventionalized chunks of words (cf. [12]), an analysis above the level of single words is an important tool in discourse analysis ([7]). By using the ngrams function, a user can calculate sequences of up to four words. The user needs to define the length of the ngram and a word and/or a part-of-speech tag that needs to be part of the ngram. E.g., a user can search for 4grams containing the word form *wir* ("we") and compare sources from media and politics, in order to identify similarities and differences in the use of the pronoun. The results will be displayed as a table or visualized as a bar chart.

Context Sensitive Analysis

In discourse analysis, but also in other fields like lexicography and language learning, context sensitive methods are crucial for analyzing the semantics of a word by its co-occurrence with other linguistic units. The workbench allows for two context sensitive modes of analysis, which differ in the size of context that is taken into account: collocation analysis and co-occurrence analysis.

- Collocation Analysis: By entering a specific word or phrase, the workbench will calculate words (so called *collocates*, cf. [13]) that occur significantly often within the immediate context of the given search string. The size of the context window can be adjusted individually, ranging from one to ten words to the right and left of the given search word, respectively. Log Likelihood is used as a measure of statistical association. Collocates can be either displayed as a table, as a bar chart or as a treemap. Collocations are especially useful to analyse the meaning of words in a given discourse, since meaning is mainly constructed by a word's immediate context.
- Co-occurrence Analysis: In contrast to the previous function, users can also identify words that correlate with a given word on a *textual level*. We use the term co-occurrence analysis to distinguish this approach from the classical window approach described for collocations. Pearson correlation is used as a statistical measure. Co-occurring words (i.e. words often appearing in the same text) can be either visualized in a bar chart or in a

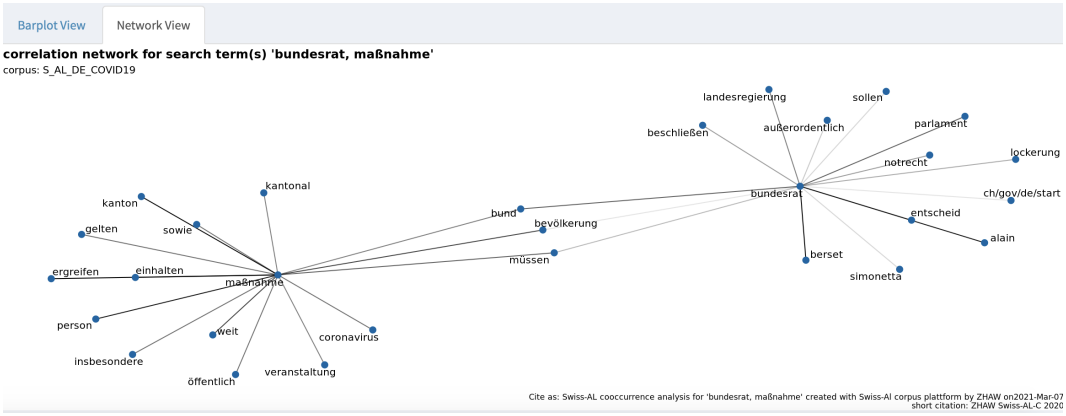


Figure 2 Co-occurrence analysis: network view. For the words *Maßnahme* ("measure") and *Bundesrat* ("Federal Council") the fifteen most correlating words on a textual level (i.e. co-occurrences) are visualized as a network. The two words share three co-occurrences (*Bund* "federation", *Bevölkerung* "population", and *müssen* "need to"), indicating a discoursive association between both words.

network. A network visualization is especially useful when co-occuring words of more than one given word should be displayed to reveal associations within a discourse (cf. Figure 2).

Keyword Analysis

Keyword analysis is one of the most established methods in corpus linguistic discourse analysis since it identifies typical vocabulary for specific discourses (or sub-discourses). The workbench allows (1) the comparison of specific years for the whole corpus (e.g., by comparing the vocabulary of 2019 with that of 2020) or (2) the comparison of specific actors for specific years (e.g., by comparing the vocabulary of a newspaper for 2019 with the same newspaper's vocabulary for 2020).

Topic Modeling

For all corpora on the workbench, separate topic models are available which can be used to get an overview over the thematic structure of the corpus. The models are precalculated by using an LDA algorithm with a prior removal of stopwords [5].[1]. Users can choose between a tabular view (showing the top 25 word of each topic) and an interactive, web-based visualization (LDAvis, [27]) to get an overview over all topics in the model and their distribution in the corpus. Furthermore, the development of topics over time can be visualized as line graphs, in order to see wether a topic is especially prominent at certain points in time.

Word Embeddings

Semantic vector space models [19] have recently become of interest in domains outside NLP. E.g., [8] shows the potential of word embeddings for the data-driven reconstruction of narrations in texts and for the analysis of public discourse. The workbench provides access to a variety of word embedding models based on the *word2vec* algorithm introduced by [22]. Models can be visualized with TensorBoard[2].

[1] Topic models were precalculated with the R wrapper for the machine learning software *Mallet* [23, 20].
[2] `https://www.tensorflow.org/`

In discourse analysis, word embeddings are especially useful for identifying semantically related words which refer to an overarching concept. E.g., users interested in the discoursive construction of fear in COVID-19 discourses could start by identifying words semantically related to *Bedrohung* ("threat") (a word from which we know that it is related to the concept of fear). The word embedding model for the Covid-19 corpus would on the one hand reveal expectable next neighbors like *Angst* ("fear") and *Panik* ("panic"), but also words that one might not think of initially but that are connected with the concept of fear in Covid-19 discourse (e.g., *Trauma* ("trauma")).

5 Conclusion

We introduced the Swiss-AL workbench as a tool for discourse analysis with a strong focus on the visualization of aggregated data and the combination of traditional corpus methods and recently developed machine and deep learning methods. The workbench is designed for a rather heterogeneous audience, i.e. researchers, practitioners and students. As such, it complements existing tools for corpus linguistic analysis. Consequently, the possibility of importing other corpora is not implemented at the moment since this would require corpus and computer linguistic expertise on the part of the user (e.g., preparing an annotated XML version of the corpus or precalculating a topic model). This scenario does not match with the expertise and needs of the intended audience of the workbench.

Next steps include the further development of the individual modes of analysis (e.g., by providing different statistical measures for keyword and collocation analysis) and the presentation of use cases and exemplary discourse analyses in order to give a user an even better understanding of how to apply corpus data to discourse analytical questions.

References

1 L. Anthony. Antconc: design and development of a freeware corpus analysis toolkit for the technical writing classroom. In *IPCC 2005. Proceedings. International Professional Communication Conference, 2005.*, pages 729–737, 2005. doi:10.1109/IPCC.2005.1494244.

2 Paul Baker. *Using Corpora in Discourse Analysis*. Continuum, London, New York, 2006.

3 Berlin-Brandenburgischen Akademie der Wissenschaften. DWDS – Digitales Wörterbuch der deutschen Sprache. Das Wortauskunftssystem zur deutschen Sprache in Geschichte und Gegenwart.

4 Andreas Blaette. *polmineR: Verbs and Nouns for Corpus Analysis*, 2020. R package version 0.8.2. doi:10.5281/zenodo.4042093.

5 David M. Blei. Probabilistic topic models. *Communications of the ACM*, 55(4):77, 2012. doi:10.1145/2133806.2133826.

6 Vaclav Brezina, P. Weill-Tessier, and A. McEnery. *#LancsBox*, 2020. v. 5.x.

7 Noah Bubenhofer. *Sprachgebrauchsmuster. Korpuslinguistik als Methode der Diskurs- und Kulturanalyse*. Number 4 in Sprache und Wissen. De Gruyter, Berlin, New York, 2009.

8 Noah Bubenhofer, Selena Calleri, and Philipp Dreesen. Politisierung in rechtspopulistischen Medien: Wortschatzanalyse und Word Embeddings. *Osnabrücker Beiträge zur Sprachtheorie (OBST)*, 95:211–241, 2019.

9 Winston Chang, Joe Cheng, JJ Allaire, Carson Sievert, Barret Schloerke, Yihui Xie, Jeff Allen, Jonathan McPherson, Alan Dipert, and Barbara Borges. *shiny: Web Application Framework for R*, 2021. R package version 1.6.0. URL: https://CRAN.R-project.org/package=shiny.

10 Nils Diewald, Michael Hanl, Eliza Margaretha, Joachim Bingel, Marc Kupietz, Piotr Bański, and Andreas Witt. KorAP architecture – diving in the Deep Sea of Corpus Data. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3586–3591, Portorož, Slovenia, 2016. European Language Resources Association (ELRA). URL: https://www.aclweb.org/anthology/L16-1569.

11 Philipp Dreesen and Julia Krasselt. Exploring and analyzing linguistic environments. In François Cooren and Peter Stücheli-Herlach, editors, *Handbook of Management Communication*, number 16 in Handbooks of Applied Linguistics. De Gruyter, Berlin, Bostom, to appear. doi:10.1515/9781501508059-021.

12 Britt Erman and Beatrice Warren. The idiom principle and the open choice principle. *Text*, 20(1):29–62, 2000.

13 Stefan Evert. Corpora and collocations. In Anke Lüdeling and Merja Kytö, editors, *Corpus Linguistics. An International Handbook*, pages 1212–1248. De Gruyter, Berlin, 2008.

14 Stefan Evert and Andrew Hardie. Twenty-first century Corpus Workbench: Updating a query architecture for the new millennium. In *Proceedings of the Corpus Linguistics 2011 Conference*, Birmingham, 2011.

15 Andrew Hardie. CQPweb – Combining Power, Flexibility and Usability in a Corpus Analysis Tool. *International Journal of Corpus Linguistics*, 17(3):380–409, 2012. doi:10.1075/ijcl.17.3.04har.

16 Adam Kilgarriff, Vít Baisa, Jan Bušta, Miloš Jakubíček, Vojtěch Kovář, Jan Michelfeit, Pavel Rychlý, and Vít Suchomel. The Sketch Engine: Ten years on. *Lexicography*, 1(1):7–36, 2014. doi:10.1007/s40607-014-0009-9.

17 Julia Krasselt, Philipp Dreesen, Matthias Fluor, Cerstin Mahlow, Klaus Rothenhäusler, and Maren Runte. Swiss-AL: A Multilingual Swiss Web Corpus for Applied Linguistics. In *Proceedings of the 12th Language Resources and Evaluation Conference (LREC)*, pages 4138–4144, Marseille, France, 2020.

18 Leibniz-Institut für Deutsche Sprache. COSMAS I/II (Corpus Search, Management and Analysis System). URL: https://cosmas2.ids-mannheim.de/.

19 Alessandro Lenci. Distributional Models of Word Meaning. *Annual Review of Linguistics*, 4(1):151–171, 2018. doi:10.1146/annurev-linguistics-030514-125254.

20 Andrew Kachites McCallum. *MALLET: A Machine Learning for Language Toolkit*, 2002. URL: http://mallet.cs.umass.edu.

21 Tony McEnery and Andrew Hardie. *Corpus Linguistics: Method, Theory and Practice*. Cambridge Textbooks in Linguistics. Cambridge University Press, Cambridge, New York, 2012.

22 Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic Regularities in Continuous Space Word Representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia, 2013. Association for Computational Linguistics.

23 David Mimno. *mallet: A wrapper around the Java machine learning tool MALLET*, 2013. R package version 1.0. URL: https://CRAN.R-project.org/package=mallet.

24 Paul Rayson. From key words to key semantic domains. *International Journal of Corpus Linguistics*, 13(4):519–549, 2008. doi:10.1075/ijcl.13.4.06ray.

25 Jan Oliver Rüdiger. CorpusExplorer, 2018. URL: http://corpusexplorer.de.

26 Mike Scott. Developing wordsmith. *International Journal of English Studies*, 8(1):95–106, 2008.

27 Carson Sievert and Kenneth Shirley. LDAvis: A method for visualizing and interpreting topics. In *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*, pages 63–70, Baltimore, Maryland, USA, 2014. Association for Computational Linguistics. doi:10.3115/v1/W14-3110.

28 Jürgen Spitzmüller and Ingo H. Warnke. *Diskurslinguistik. Eine Einführung in Theorien und Methoden der transtextuellen Sprachanalyse*. De Gruyter, Berlin, Boston, 2011.

29 Sascha Wolfer, Alexander Koplenig, Frank Michaelis, and Carolin Müller-Spitzer. *cOWIDplus Viewer*, 2020. URL: https://www.owid.de/plus/cowidplusviewer2020/.

APiCS-Ligt: Towards Semantic Enrichment of Interlinear Glossed Text

Maxim Ionov
Applied Computational Linguistics Lab, Goethe University Frankfurt, Germany

Abstract

This paper presents APiCS-Ligt, an LLOD version of a collection of interlinear glossed linguistic examples from APiCS, the Atlas of Pidgin and Creole Language Structures. Interlinear glossed text (IGT) plays an important role in typological and theoretical linguistic research, especially with understudied and endangered languages: It provides a way to understand linguistic phenomena without necessarily knowing the source language which is crucial for these languages since native speakers are not always easily accessible.

Previously, we presented Ligt, RDF vocabulary created for representing interlinear glosses in text segments. In this paper, we present our conversion of the APiCS IGT dataset into this model and describe our efforts in linking linguistic annotations to an external ontology to add semantic representation.

2012 ACM Subject Classification Information systems → Graph-based database models; Computing methodologies → Language resources; Computing methodologies → Knowledge representation and reasoning

Keywords and phrases Linguistic Linked Open Data (LLOD), less-resourced languages in the (multilingual) Semantic Web, interlinear glossed text (IGT), data modeling

Digital Object Identifier 10.4230/OASIcs.LDK.2021.27

Supplementary Material *Software (Source Code and Dataset)*: `https://github.com/acoli-repo/ligt/tree/master/stable/apics/`
Dataset: `https://doi.org/10.5281/zenodo.5155753`

Funding This work was funded by the project "Prêt-à-LLOD" within the European Union's Horizon 2020 research and innovation programme under grant agreement no. 825182, as well as the project "Linked Open Dictionaries" (LiODi), funded within the eHumanities program of the German Ministry of Education and Science (BMBF, 2015-2021).

1 Background

Linguistic examples with interlinear glossing, be that texts or elicitations, are crucial for linguistic research since they provide a way to understand linguistic structures in languages researchers do not know. Both for exploring language material and to provide proof of a claim, they accompany linguistic research on all stages.

This data may consist of any number of layers: free translation, word-by-word translation, grammatical meaning of morphemes, transliteration, etc. Some layers has morpheme-by-morpheme correspondence between each other, e.g. morpheme segmentation and grammatical meaning of morphemes. Consider the following example in Gurindji Kriol language:[1]

(1) Jambala dei meikim nyawanginyima.

Jambala dei meik-im nyawa-nginyi-ma.
somebody 3PL.SBJ make-TR this-ABL-TOP

"Some people make it out of this one."

[1] For source data, attribution and more information see `https://apics-online.info/sentences/72-35`.

In this example, there are two layers without morpheme alignment ("baseline" and "free translation") and the other two are aligned. The list of layers is not restricted, but there are guidelines, Leipzig Glossing Rules (LGR) [4].

In our previous work [2], we presented Ligt, RDF-native vocabulary capable of representing *structure* of IGT and demonstrated how it could be used to model data produced by widely-used tools for field linguistics: Toolbox and FLEx (based on our research before that [3]). Since then, the vocabulary was also used to represent a massive typologically diverse dataset based on language archive data [14].

While providing a shared model for different source formats increase interoperability between *formats*, i.e. allowing to query over data produced with different tool sets, it does not save against variability of annotations. LGR provide a list of commonly used abbreviations for grammatical categories (e.g. ABL for Ablative case), but this list is neither full nor universally used, and both these reasons lead to mismatches in tags across different datasets. Usually there is a list of abbreviations either in a book or attached to the dataset,[2] and this could be used for disambiguating the labels. However, these are still *labels* (strings), not *categories*.[3] In order to provide semantics for these labels, we create a mapping linking the labels with external ontologies.

The rest of the paper is structured as follows: in Section 2 we describe the source dataset, Section 3 briefly presents the Ligt data model. Sections 4 and 5 are devoted to the conversion and the linking respectively. Finally, Section 6 concludes the paper also pointing out future work.

2 APiCS

The Atlas of Pidgin and Creole Languages [13] is an online database[4] with linguistic information on 76 pidgin and creole languages of the world. This information includes grammatical and lexical features of these languages, collection of references, grammatical surveys. Most importantly for this paper, this database contains a collection of linguistic examples with interlinear glossing (18 526 in total). These examples are of different nature: naturalistic spoken, written or translated, constructed by a linguist, a native speaker, etc. Some of these examples are augmented with speech recordings.

APiCS data model is based on Cross-Linguistic Data Formats (CLDF) [7] which is employed by several typological databases due to its convenience in installation and usage. The model is based on the W3C standard "Model for Tabular Data on the Web" [10] which, in turn, is a dialect of JSON-LD, which lead to the database structure having semantic annotations. Examples are connected to additional information, such as presence of certain grammatical features, but their internal structure is stored as strings, without connections to structured information, e.g. tables of features, meaning that it is not possible to query these examples for grammatical categories in an easy way.

In order to preserve the original annotations, but add internal structure, we decided to use APiCS sentence identifiers as identifiers in Ligt annotation and add `owl:sameAs` links from Ligt sentence fragments back to APiCS.

[2] `https://github.com/cldf-datasets/apics/blob/master/cldf/glossabbreviations.csv`
[3] See also [14, Section 4.3].
[4] `https://apics-online.info/`

3 Ligt

Ligt vocabulary is grounded in three well-established vocabularies: Dublin Core [17], NIF [9] and WebAnnotation [15]. Since this paper focuses on the application of the vocabulary, and not on its definition, we will list only its key aspects here. For a more in-depth description, related work and a survey on alternative representations for IGT, see [2]. Below are some key aspects and new additions:

- The central element is `ligt:Document`, a subclass of `dc:Dataset`. Objects of this class can have multiple pointers to texts and sentences. Previously, the model was limited to collections of texts via the `ligt:hasText` property with an object of type `ligt:InterlinearText` or (`dc:Text`). Since a large amount of IGT data, including APiCS database consists of elicitations or at least of sentences not organized into bigger elements, we have introduced a new property: `hasUtterances` with an object of type `ligt:InterlinearCollection`.
- `ligt:InterlinearCollection` consists of one or more utterances. Some datasets logically consist of independent (or weakly dependent) parts which can be modeled with a single document having multiple `hasUtterances` properties pointing to different `ligt:InterlinearCollection` instances.
- As with text, Using NIF predicate `nif:subString` it is possible to split a text or a interlinear collection into smaller parts: `ligt:Paragraph` or `ligt:Utterance`. `ligt:Utterance` roughly corresponds to a sentence or an elicitation.[5]
- To represent layers, we introduced a class `ligt:Tier` and two subclasses: `ligt:WordTier` and `ligt:MorphTier` which should correspond to sequences of words and morphs, respectively. Tiers in Ligt must consist of elements on the same level of granularity (e.g. words with words vs. morphemes with morphemes).
- Both `ligt:Word` and `ligt:Morph` are subclasses of `ligt:Item` and are objects of a property `ligt:item` for the word and morph tiers, respectively.
- An instance of a tier (a sentence or a word) should have a property `ligt:item` that points to its smaller components. Components within one tier must be connected by a property `ligt:next`.
- Data properties can be added to an item depending on the data (e.g. translation)
- Finally, for compatibility with FLEx data, we keep subclasses of `ligt:Morph` for representing prefixes, suffixes, stems and enclitics.

The data model (excluding metadata) is illustrated in Fig. 1.

4 Conversion

4.1 Conversion details

APiCS example sentences are stored in a CSV table which conforms to a schema[6] describing which layers can be in the data, whether they are required and if there is a separator symbol for the data (e.g. morpheme line).[7] Each row corresponds to a separate sentence so the

[5] Splitting `ligt:InterlinearCollection` into paragraphs might seem strange, but this can, in fact, lead to a nicer modeling: if a group of elicitations is not big enough to be a `ligt:InterlinearCollection` but there need to be some grouping (e.g. subsection in a grammar or a group of examples related to a single phenomenon).

[6] `http://cldf.clld.org/v1.0/terms.rdf#ExampleTable`

[7] Separators were not present in the previous release of the data so initially we split the data during conversion heuristically.

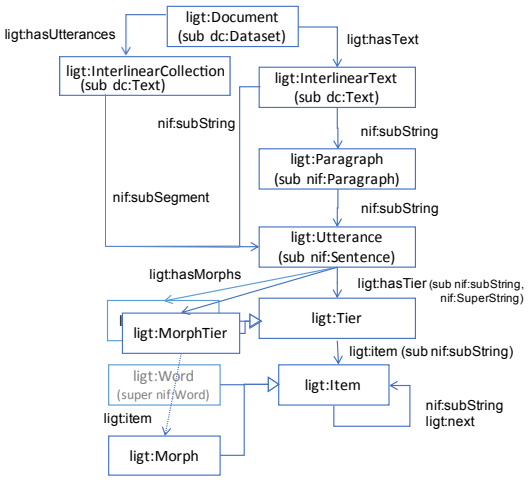


Figure 1 Ligt data model.

conversion process was limited to creating triples with the dataset metadata, adding triples for each sentence, and creating layers for sequences of words for each sentence and for sequences of morphs for each word. The resulting structure is the following:

- Dataset-specific metadata: bibliographic citation
- One `ligt:Document` for all the sentence
- A single `ligt:InterlinearCollection` for all the sentences
- Metadata for each `ligt:Utterance` (sentence): language code, comment, `owl:sameAs` with a link to APiCS[8]
- 3 tiers for each sentence: phrase, words, and morphs
- Original text as an `rdfs:label`, translation as an object of `ligt:translation`,[9] and a comment as an `rdfs:comment`
- For every morph: original text in a `rdfs:label`, gloss marker in a `ligt:gloss`

An excerpt from the converted data is illustrated on Fig. 2.[10]

4.2 Querying

Even after purely structural conversion, without adding semantic information to linguistic categories, it is possible to perform qualitative and quantitative analysis on the dataset. Doing corpus analysis on RDF datasets is beyond the scope of this paper, so we will just demonstrate some exploratory queries.

First query returns grammatical markers which are found in the most number of languages:

[8] In this version of the conversion, we do not add attribution and provenance information for each sentence, but it is easily retrievable since there is a link to the original example record in APiCS.

[9] Here we do not model free translation as a separate `Tier`, but creating a separate tier for it would be a possible design decision, and in fact, a single SPARQL update can be used to convert between the two.

[10] Full resolution and more diagrams can be found at `https://github.com/acoli-repo/ligt/tree/master/stable/apics/diagrams/`

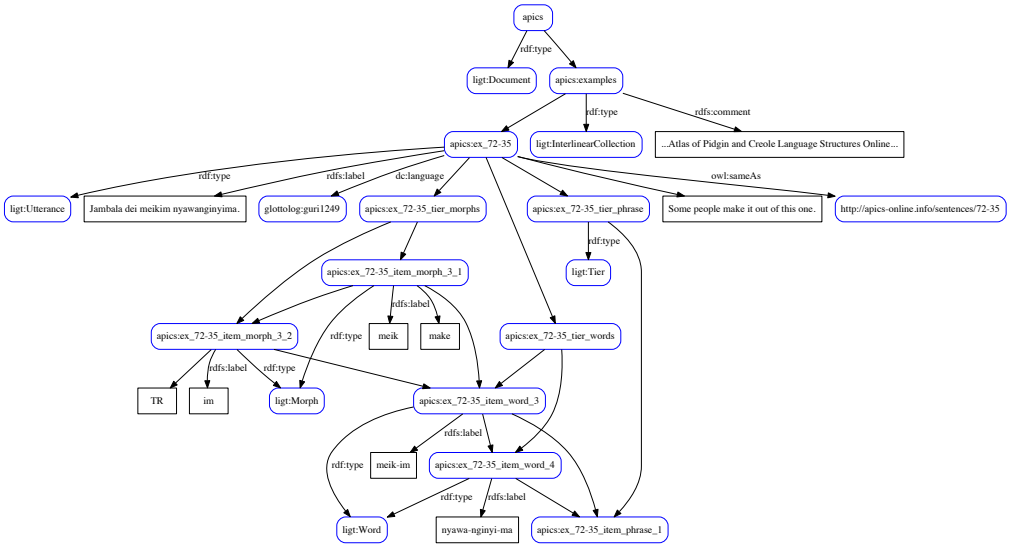


Figure 2 APiCS-Ligt example.

```
PREFIX ligt: <http://purl.org/ligt/ligt-0.2#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT (COUNT(?lang) as ?n_lang) ?val
WHERE {
  ?morph ligt:gloss ?val ;
         rdfs:label ?label .

  BIND(LANG(?label) as ?lang)
  FILTER(?val = UCASE(?val) && ?lang != '')
} GROUP BY ?val ORDER BY DESC(?n_lang)
```

Marker	#
3SG	2650
1SG	2397
NEG	1400
2SG	1306
PST	1099

We can also look for more typologically interesting questions. For example, it might be possible to see the morphosyntactic alignment strategies that exist in languages of the dataset.[11] An easy approximation of this would be to look at the presence of Accusative and Ergative grammatical markers in language data:

```
PREFIX ligt: <http://purl.org/ligt/ligt-0.2#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT ?case ?lang
WHERE {
VALUES ?case { "ACC"@en "ERG"@en }
?morph ligt:gloss ?case;
       rdfs:label ?label .
  BIND (LANG(?label) AS ?lang)

  FILTER(?lang != '')
}
```

Case	Language
ACC	idb-x-dama1278
ACC	mcm-x-mala1533
ACC	pga-x-suda1237
ACC	sci-x-sril1245
ACC	mue-x-medi1245
ERG	gjr-x-guri1249

This query points to an obvious problem: it is necessary to list all the labels for grammatical cases, we can not query for all possible sets of them. In order to be able to do so, we need to map the labels to some external source, to augment string labels with linguistic categories.

[11] Typologically, there are tendencies to have certain combination of case markers on subjects and objects. Most notably, Nominative-accusative and Ergative-absolutive types. [5]

5 Linking

5.1 Mapping to ontologies

There has been a debate regarding universality and cross-applicability of linguistic categories [8]. While this is, undoubtedly, an important topic, and we carefully agree with the premise that sparked the debate, having linguistic categories such as parts of speech as an approximation is extremely helpful for practical reasons. Nevertheless, we find it important not to overgeneralize and this was one of the concerns in choosing the source we could map APiCS annotation to.

There is a variety of community-maintained repositories of annotation terminology evolved during the 2000s, which aimed to replace annotation standards by collecting and defining categories without requiring them to be disjoint. Exemplary repositories developed at this time include ISOcat [11], developed with a specific focus on language technology, and the General Ontology of Language Description [6], developed with a specific focus on language documentation.

Another repository designed to be flexible and non-reductionist is OLiA, Ontologies of Linguistic Annotation [1]. In its conception, OLiA aimed to address what could be called the "standardization gap" of linguistic annotation. That means that a consistent standardization of linguistic annotation would either have to neglect language specific characteristics (cf. Universal Dependencies tagset), or constantly grow in complexity with every new language added to it. OLiA is modular, and allows users formalize their annotation schemes and to link them with reference concepts. This approach suits our task very well given that:

- The set of markers in APiCS is quite extensive – matched against standard Leipzig Glossing Rules list of abbreviations, we got less than a quarter of the markers matched (23.54%). The list of glosses that is distributed with the dataset has 267 abbreviations.
- Annotations in the dataset are *morpheme markers*, they does not necessarily correspond to grammatical categories: reduplication, oblique stem and agreement are present in the dataset as morpheme values, but they do not directly correspond to a grammatical category (which could be, e.g. an intensifier in case of reduplication).
- One of the modules is a morpheme inventory converted from the UniMorph project [16] which links OLiA Reference Model classes to UniMorph morpheme inventory.

Additionally to OLiA, we decided to map morpheme labels to the morpheme inventory in the MMoOn Core ontology [12]. This ontology was created to provide a shared semantic model for morphological information, which is precisely our goal. In the core of this ontology there is a language-independent collection of morphemes with their labels and description, which we also referenced to enrich our morpheme annotations.

By matching tags and their description we were able to map 123 unique labels, 81 with OLiA ontologies and 91 with MMoOn. For each mapping, we added an additional statement to the dataset:

```
<http://mmoon.org/core/Ablative> apics:hasValue "ABL"@en .
<http://mmoon.org/core/Absolutive> apics:hasValue "ABS"@en .
<http://mmoon.org/core/Accusative> apics:hasValue "ACC"@en .
<http://mmoon.org/core/Active> apics:hasValue "ACT"@en .
<http://mmoon.org/core/Adjective> apics:hasValue "ADJ"@en .

<http://purl.org/olia/unimorph.owl#ABL> apics:hasValue "ABL"@en .
<http://purl.org/olia/unimorph.owl#ABS> apics:hasValue "ABS"@en .
<http://purl.org/olia/unimorph.owl#ACC> apics:hasValue "ACC"@en .
<http://purl.org/olia/unimorph.owl#ACT> apics:hasValue "ACT"@en .
<http://purl.org/olia/unimorph.owl#ADJ> apics:hasValue "ADJ"@en .
```

In our future work we will analyze which labels did not map and whether it is possible to find mappings for them.

5.2 Querying

Now that we have semantic value behind some of the tags, we can query using this additional knowledge. Query below groups all the case markers encountered in sentences in each language:[12]

```
PREFIX ligt: <http://purl.org/ligt/ligt-0.2#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX apics: <http://purl.org/liodi/ligt/apics/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX mmcore: <http://mmoon.org/core/>
PREFIX unimorph: <http://purl.org/olia/unimorph.owl#>

SELECT (GROUP_CONCAT(DISTINCT ?case; SEPARATOR=", ")
        AS ?cases) ?lang
WHERE {
  ?morph ligt:gloss ?case ;
         rdfs:label ?label .
  ?tag apics:hasValue ?case .
  { ?tag rdfs:subClassOf+ mmcore:Case . }
  UNION
  { ?tag rdfs:subClassOf+ unimorph:Case . }


  BIND(LANG(?label) as ?lang)
  FILTER(?lang != '')
} GROUP BY ?lang
```

Cases	Language code
LOC, COM, INS, DAT, TEMP	rop-x-krio1252
LOC, INS, ABL, BEN, ALL, ACC, GEN	mue-x-medi1245
LOC, COM, INS, MOD, ABL, ALL, DAT, ERG	gjr-x-guri1249
INS	gcf-x-guad1242
LOC, VOC, GEN	kcn-x-nubi1253
LOC, VOC, MOD	jam-x-jama1262
COM, INS, VOC	pov-x-uppe1455
LOC, VOC, MOD	srm-x-sara1340
LOC	fpe-x-fern1234
LOC, COM, INS	bah-x-baha1260
VOC	lou-x-loui1240
...	

6 Conclusion

In this paper we presented APiCS-Ligt, an RDF edition of interlinear glossed linguistic examples from the Atlas of Pidgin and Creole Languages. Our conversion remains linked with the original dataset therefore preserving all additional information such as bibliographical references or linguistic features, but at the same time adding for linguistic examples both a structural level and a layer of semantics, providing more interpretability to linguistic annotations and interoperability with another resources with linguistic annotations.

We showed that such semantic linking can be problematic due to both practical (ambiguity of markers) and theoretically-motivated (differences in definitions of linguistic categories) reasons which might be improved if linguists were more involved in the data modeling and data standardization stages.

[12] We give only an excerpt of the results in the table below.

In the future we are planning to go beyond APiCS IGT data to other sources of IGT to see how transferable are the solutions that we came up with. We also plan on publishing a Python module aimed at combining in one place all previously developed procedures for importing and exporting Ligt format and add functionality for working with Ligt data.

The dataset and the code to reproduce the conversion is available at `https://github.com/acoli-repo/ligt/tree/master/stable/apics/`.

References

1 C. Chiarcos and M. Sukhareva. OLiA – Ontologies of Linguistic Annotation. *Semantic Web Journal*, 518:379–386, 2015.

2 Christian Chiarcos and Maxim Ionov. Ligt: An LLOD-Native vocabulary for representing Interlinear Glossed Text as RDF. In *2nd Conference on Language, Data and Knowledge (LDK 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2019.

3 Christian Chiarcos, Maxim Ionov, Monika Rind-Pawlowski, Christian Fäth, Jesse Wichers Schreur, and Irina Nevskaya. LLODifying linguistic glosses. In *Proceedings of Language, Data and Knowledge (LDK-2017)*, Galway, Ireland, June 2017.

4 Bernard Comrie, Martin Haspelmath, and Balthasar Bickel. The Leipzig Glossing Rules: Conventions for interlinear morpheme-by-morpheme glosses. `https://www.eva.mpg.de/lingua/pdf/Glossing-Rules.pdf`, 2008.

5 Robert MW Dixon. *Ergativity*. Cambridge University Press, 1994.

6 S. Farrar and D. T. Langendoen. An OWL-DL implementation of GOLD: An ontology for the Semantic Web. In A. Witt and D. Metzing, editors, *Linguistic Modeling of Information and Markup Languages: Contributions to Language Technology*. Springer, Dordrecht, Netherlands, 2010.

7 Robert Forkel, Johann-Mattis List, Simon J Greenhill, Christoph Rzymski, Sebastian Bank, Michael Cysouw, Harald Hammarström, Martin Haspelmath, Gereon A Kaiping, and Russell D Gray. Cross-linguistic data formats, advancing data sharing and re-use in comparative linguistics. *Scientific data*, 5(1):1–10, 2018.

8 Martin Haspelmath. Pre-established categories don't exist: Consequences for language description and typology. *Linguistic typology*, 11(1), 2007.

9 Sebastian Hellmann, Jens Lehmann, Sören Auer, and Martin Brümmer. Integrating NLP using Linked Data. In *Proc. 12th International Semantic Web Conference, 21-25 October 2013*, Sydney, Australia, 2013. also see `http://persistence.uni-leipzig.org/nlp2rdf/`.

10 Gregg Kellogg and Jeni Tennison. Model for tabular data and metadata on the web. W3C recommendation, W3C, 2015. `https://www.w3.org/TR/2015/REC-tabular-data-model-20151217/`.

11 M. Kemps-Snijders, M. Windhouwer, P. Wittenburg, and S. E. Writh. ISOcat: remodelling metadata for language resources. *International Journal of Metdata, Semantics and Ontologies*, 4(4):261–276, 2009.

12 Bettina Klimek, Markus Ackermann, Martin Brümmer, and Sebastian Hellmann. Mmoon core-the multilingual morpheme ontology. *Semantic Web Journal*, 2020.

13 Susanne Maria Michaelis, Philippe Maurer, Martin Haspelmath, and Magnus Huber, editors. *APiCS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig, 2013. URL: `https://apics-online.info/`.

14 Sebastian Nordhoff. Modelling and annotating interlinear glossed text from 280 different endangered languages as linked data with ligt. In *Proceedings of the 14th Linguistic Annotation Workshop*, pages 93–104, 2020.

15 Robert Sanderson, Paolo Ciccarese, and Herbert Van de Sompel. Open annotation data model. Technical report, W3C Community Draft, 08 February 2013, 2013.

16 John Sylak-Glassman. The composition and use of the universal morphological feature schema (unimorph schema). *Johns Hopkins University*, 2016.

17 S. Weibel, J. Kunze, C. Lagoze, , and M. Wolf. RFC 2413 - Dublin Core metadata for resource discovery. URL `http://www.ietf.org/rfc/rfc2413.txt` (July 31, 2012), September 1998. Network Working Group.

Introducing the NLU Showroom: A NLU Demonstrator for the German Language

Dennis Wegener
Fraunhofer Institute for Intelligent Analysis and Information Systems, Sankt Augustin, Germany

Sven Giesselbach
Fraunhofer Institute for Intelligent Analysis and Information Systems, Sankt Augustin, Germany

Niclas Doll
Fraunhofer Institute for Intelligent Analysis and Information Systems, Sankt Augustin, Germany

Heike Horstmann
Fraunhofer Institute for Intelligent Analysis and Information Systems, Sankt Augustin, Germany

Abstract

We present the NLU Showroom, a platform for interactively demonstrating the functionality of natural language understanding models with easy to use visual interfaces. The NLU Showroom focuses primarily on the German language, as not many German NLU resources exist. However, it also serves corresponding English models to reach a broader audience. With the NLU Showroom we demonstrate and compare the capabilities and limitations of a variety of NLP/NLU models. The four initial demonstrators include a) a comparison on how different word representations capture semantic similarity b) a comparison on how different sentence representations interpret sentence similarity c) a showcase on analyzing reviews with NLU d) a showcase on finding links between entities. The NLU Showroom is build on state-of-the-art architectures for model serving and data processing. It targets a broad audience, from newbies to researchers but puts a focus on putting the presented models in the context of industrial applications.

2012 ACM Subject Classification Applied computing → Document management and text processing

Keywords and phrases Natural Language Understanding, Natural Language Processing, NLU, NLP, Showroom, Demonstrator, Demos, Text Similarity, Opinion Mining, Relation Extraction

Digital Object Identifier 10.4230/OASIcs.LDK.2021.28

Funding This research has been funded by the Federal Ministry of Education and Research of Germany as part of the competence center for machine learning ML2R (01IS18038B).

1 Introduction

Natural language processing (NLP) and understanding (NLU) have gained a lot of interest over the past years. In the following, we will refer to both NLP and NLU as NLU. Recent developments in the field have led to significant improvements for many use cases, enabling the usage of NLU Models in real world application. This applies especially for English, for which a lot of large annotated corpora are available. However, this leaves a gap for other languages such as German.

The goal of the NLU Showroom is to demonstrate the capabilities of NLU models for the German language in order to raise interest from people outside of the scientific domain for integrating NLU models in their applications. The demonstrators provided with the NLU Showroom should serve as showcases for the capabilities of state-of-the-art NLU models and their limitations for a variety of tasks. In the current version we provide demonstrators for a) comparing how different word representation models capture semantic similarity of words when trained on the same corpus, i.e. the German Wikipedia b) Comparing how different sentence representation models interpret sentence similarity when trained on the same corpus,

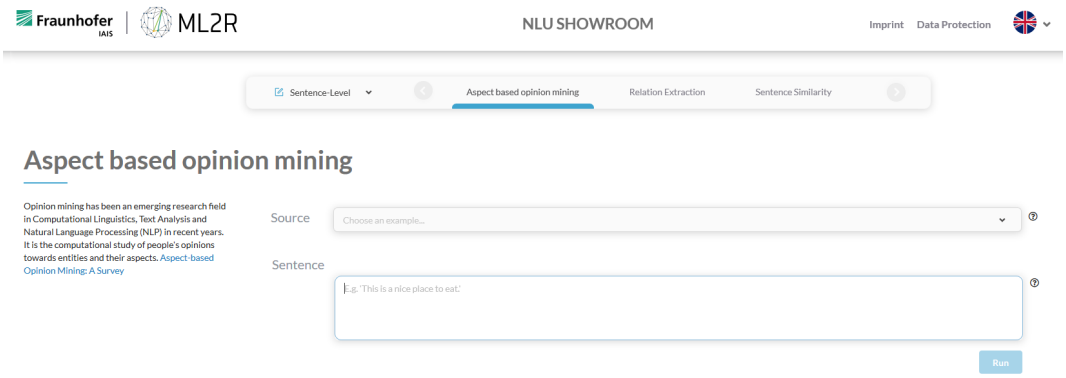


Figure 1 The interface of the NLU Showroom lets the user navigate through different demos and lets them interactively explore NLU model capabilities and differences. The users can enter their own prompt or select example inputs.

also Wikipedia c) showcasing how multiple state-of-the-art BERT [9] models can be used to analyze reviews and extract and link aspects and opinions d) showcase how state-of-the-art relation extraction models, extract relations for preselected entities.

The heart of the NLU Showroom is a web-based frontend, which allows the user to interactively explore NLU models in multiple demonstrators (see Figure 1). At it's core, the NLU Showroom has been built on a strong architectural base, so that it can easily be extended with additional models based on PyTorch and TensorFlow as well as other non-neural models. In the following we will refer to NLU tasks such as named entity recognition as "tasks", the models used to solve the tasks as "models" and the interactive demo of a model regarding a task as "demonstrator".

2 Related platforms and demos

A handful of NLU demo platforms already exist. Most of them are demonstrating natural language analysis packages and appeal to experts in the field of NLU rather than to people who want to understand how to utilize NLU in their applications.

A very recent addition is Stanfords Stanza [21]. It is a Python natural language analysis package which is also presented in an online demo [5]. The demo includes the models provided in the package, namely for Part-of-Speech Tagging, Lemmatization, Named Entity Recognition (NER) and Dependency Parsing models for a variety of languages (including German). All of these models represent important steps in the analysis and understanding of texts. However, with the exception of NER, these models address rather low-level tasks and clearly tackle NLU practitioners. They might be too in-depth for people outside of the NLU domain. The same applies for the four demos of the python natural language toolkit (NLTK) [4]. The demos include word tokenizers, stemming in 17 languages, part of speech tagging with 22 different part of speech taggers and finally sentiment analysis, which uses text classification to determine sentiment polarity in 3 languages.

The IBM Watson NLU demo on Text Analysis [3] includes the extraction of entities, keywords, concepts and relations, the classification of sentiment, emotions and categories, and linguistic analysis of semantic roles and syntax. The demo presents prepared examples from different industrial domains and allows to enter own text or URL pointers. However, in contrast to the NLU Showroom, it provides no information about the models used for the different tasks.

The European Language Grid [22] is a platform for supporting language technologies in the European market. It aims at delivering not only datasets but also services and demos for EU official languages and EU languages without official status. The services include a variety of NLP tasks, also for the German language. However, the services seem to be rather focused on NLU practitioners and scientists than on the general public.

The same applies to the Hugging Face Transformer framework [27]. Hugging Face is an open source framework providing pretrained models for a variety of NLP tasks based on Transformer models only. It aims at data scientists and researchers and is primarily focused on the sharing of models and data.

The demonstrator the NLU Showroom shares most similarities with is the one from AllenNLP [10]. AllenNLP is an open-source natural language processing platform for building state of the art models, which also includes a demo on the functionality of NLU models [1]. It includes demonstrations for many NLU tasks, including those that the NLU Showroom aims at. However, the demos and models are only available in the English language.

3 Tasks and Models

For the initial release of the NLU Showroom we selected a variety of tasks which demonstrate how NLU models work, what they are capable of, what their limitations are and how they can be used. We designed demonstrators which easily show how different models understand natural language. The tasks in the demos are of different granularity. Some of them represent word-level tasks, while others are on a sentence basis. In addition, the tasks are of different complexity levels, e.g. single models vs several models that are combined to complete a certain task. The showroom contains short explanations of all the tasks, the models and the data they were trained on.

In the following, we will present the tasks and describe the underlying models. All models were trained on German and English data sets.

3.1 Word-Level

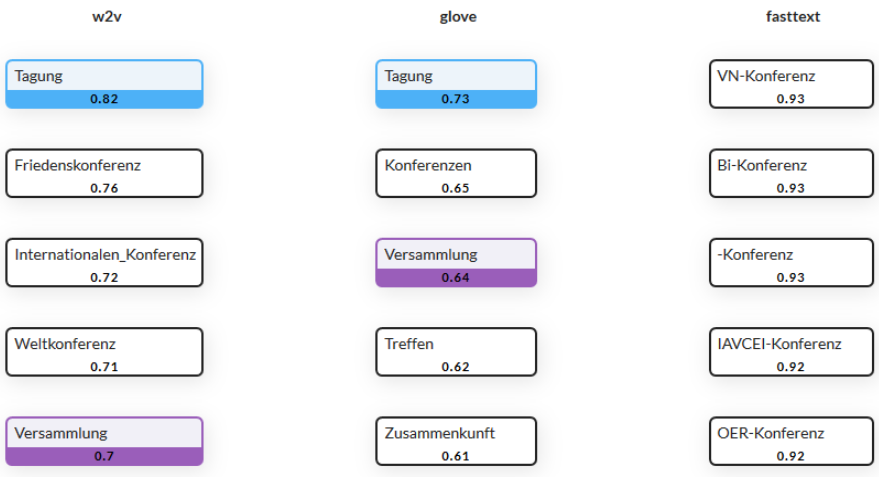


Figure 2 An example output of the word similarity task for the word “Konferenz”. The user can see how 3 models evaluate the similarity in a different way. Words that are highlighted in colours are returned by multiple models. The numbers represent the cosine similarity to the query word.

The NLU Showroom currently contains one word-level demonstrator. The demonstrator on the word-level basis shows how different word representation models capture semantic similarity when trained on the same corpus with similar hyper-parameters. We trained multiple distributed embedding models on the same pre-processed versions of the German and English Wikipedia. The models we trained are Word2Vec [15], Glove [18] and FastText [8]. We have two intentions with this demonstrator: 1) Show that these representations capture meaningful similarities and can be used e.g. to enrich search functions or ontologies 2) Show that these models capture similarities in different ways – e.g. that the influence of using character n-grams in FastText heavily influences what is considered similar when compared to Word2Vec. Reason 2) also motivates why we did not use pretrained embeddings since the data they were pre-processed with was most likely from multiple sources. For the same reason all models have been trained with roughly the same parameters. In detail, all embeddings have been trained with embedding dimensionality set to 300, a context window size of 5 and 5 negative samples. Our word similarity demo (see Figure 2) lets users input query words and phrases and explore the nearest neighbors that the models obtain. It highlights similarities and differences between the responses of the different models. An interesting observation is that despite their theoretical similarity [14] Word2Vec and Glove produce surprisingly different results. It is also easily observable that FastText focuses more on morphological similarities. In the future we will add further word representation models and let the users select which models to compare.

3.2 Sentence-Level

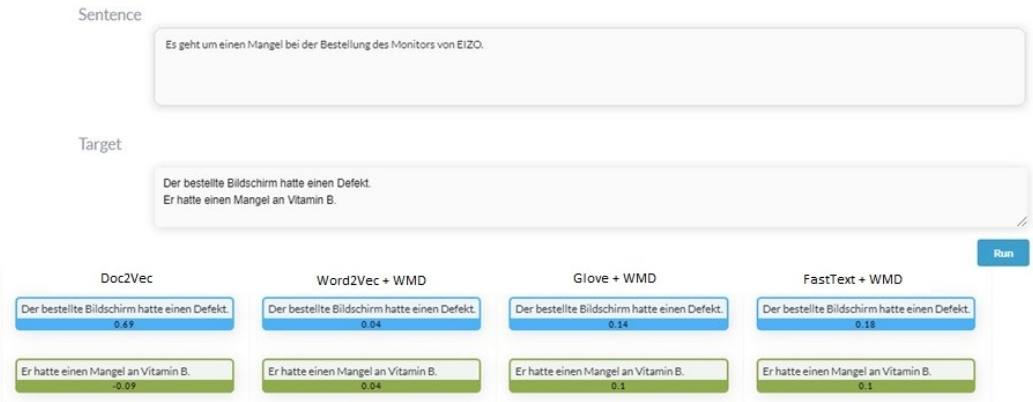


Figure 3 An exemplary output of the sentence similarity task. The user can enter a source sentence and several target sentences. As result we show the similarities of the target sentences to the source sentence computed by the different models. A higher number means greater similarity. Across models the same sentence is highlighted in the same color for easier comparison.

In our initial release of the NLU Showroom we integrate demonstrators for 3 different sentence-level tasks. The first demo displays a sentence similarity task, similar to the word-similarity task. Users can input a source sentence and target sentences and then our models compare the similarity of the source sentence to the different target sentences. Many text based processes require the need to compare texts and identify similar texts or passages. We intend to highlight how the choice of model can influence the similarities and that the models rely on more than simple string or keyword matching. Our demonstrator currently includes 4 different models. We trained a Doc2Vec [13] model on the German and English Wikipedia

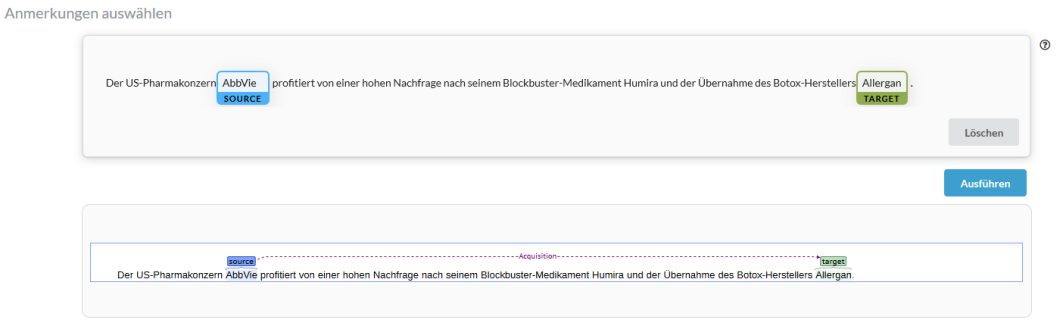


Figure 4 The output of our relation extraction and classification demonstrator. The demo requires 2 steps: 1) The user inputs a sentence 2) the user highlights the entities which should be checked for a possible relation. If a relation is detected, the name of the relation is returned.

and use the Word Mover's Distance [12] together with the 3 different word representation models mentioned in the word-level task. The demo displays the similarity rankings of the different target sentences to the source sentence. Figure 3 shows an example call of the sentence similarity task. Notably all models understand that target sentence 1 is closer to the source sentence, even though target sentence 2 and the source sentence share more words. We will integrate new Transformer-based similarity measures soon.

Our second sentence-level demo, demonstrates a relation extraction and classification model. Here we use a BERT-based model [24] that was trained on the German Smartdata Corpus [23] and the English SemEval2010 Task8 data set [11]. In the demo, the user can input a sentence and specify entities which should be checked for relations (see Figure 4). The model then visualizes the relation class, if a relation has been detected.

Third, we present an aspect-based opinion mining demo. This demo actually combines three models, trained on German and English Yelp reviews. It starts by extracting aspects and opinions from the text using a BERT-based architecture [25]. Another BERT-based model is applied to compute sentiment labels. Lastly we link aspects and opinions by finding shortest paths in the dependency tree. The dependency tree is build using StanfordNLP's dependency parser [20].

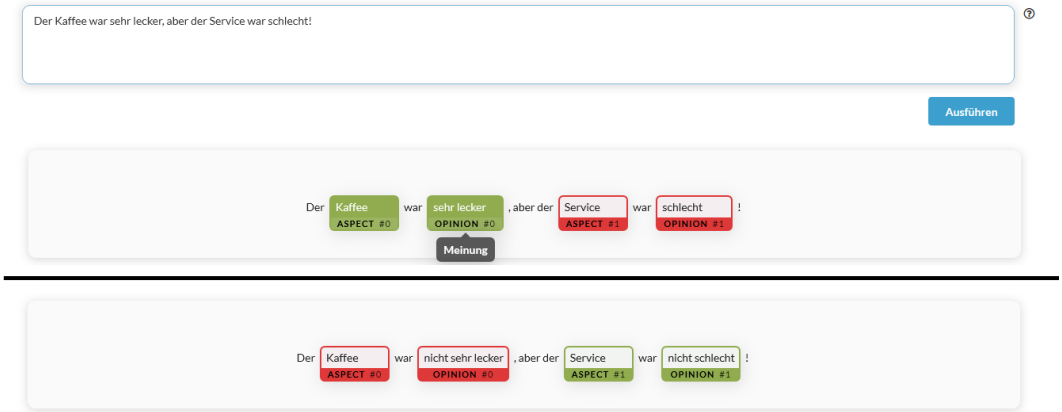


Figure 5 Exemplary outputs of the opinion mining task. The first sentence includes two different sentiments (aspect-opinion pairs). Aspects and opinions are linked via a number next to their description and links are highlighted when hovered over. Green represents positive and red negative sentiment. In the second sentence we negated the opinions.

The English models were trained on the SemEval2014-Task4 data set [19], which originally only consisted of aspect labels and their sentiments, but was extended to also annotate opinion labels [26]. For the German language we crawled and manually annotated reviews from Yelp to create a dataset, which we aim to publish in the future. The demo asks the users to enter a restaurant review and will then output the extracted aspects and opinions as well as the links between them and the according sentiments (see Figure 5). It serves as a demonstration of what is possible in automated review analysis with state-of-the-art models. Experiments with the models show that it is rather resistant to spelling mistakes, can handle negations and multiple aspect and opinion pairs. However, it fails when multiple opinions are linked to the same aspect.

4 Technical Architecture

The NLU Showroom demonstrates state-of-the art NLU models. These models are typically created with the help of modern open source frameworks, such as TensorFlow [7] and PyTorch [17]. These frameworks also already include components for serving the models. As part of TensorFlow Extended (TFX), TensorFlow Serving [16] is a serving system for machine learning models which is designed for production environments. It provides integration with TensorFlow models and can be easily extended to serve other types of models and data. TorchServe [6] is a light-weight tool for deploying and serving PyTorch models. Since the NLU Showroom is focused on quickly demonstrating latest models from ML research, we integrated these two open source frameworks in the backend of our NLU Showroom. The training process of the models is not explicitly addressed by our architecture and is performed offline in a GPU computing cluster.

In detail, the NLU Showroom consists of a frontend and several backend components. The frontend is a web application that is based on frontend frameworks such as Node.js and React. It includes the actual web pages, the controls to interact with the models and the different visualizations for the model results. The frontend communicates with the backend part via REST services. The backend includes several components which are mainly responsible for serving the models. As we already stated, we use TensorFlow Serving [16] and TorchServe [6] for serving models from the different frameworks. In addition, a webserver that serves as wrapper for the model serving components is responsible for transforming the model results into visualization content and also for wrapping several models for a single task, e.g., for the task on aspect based opinion mining where 3 models are combined.

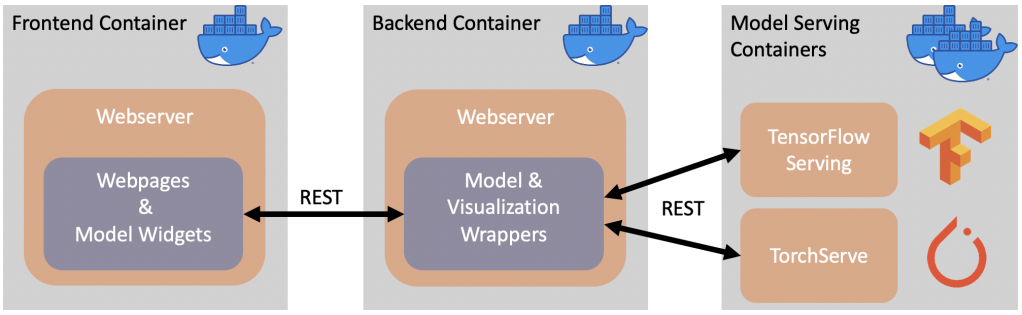


Figure 6 The technical architecture of the NLU Showroom.

The whole architecture is based on Docker [2] in order to easily isolate the different components. This allows us to take standard open source components as TensorFlow Serving and TorchServe without any modifications, and isolate our wrapper and glue code in a single

separate component (the web server). The containerization via Docker also allow us to later deploy the system into an auto-scaling production environment. Figure 6 gives an overview over the technical architecture.

New models can be integrated in an easy way. They just have to be deployed into the TorchServe or the TensorFlow Serving container. If the new model refers to an existing task, there are just minor adaptions needed on the wrapper backend component to forward the model output to the frontend. If it refers to a new task, there is the need to develop a suitable widget for it. In addition, the wrapper needs to be extended for tasks that include several models or build on models that are not deployed into TorchServe or TensorFlow Serving but need custom integration. Currently, the set of tasks and models is curated by the project team. In the future we might open the NLU Showroom to further contributors.

5 Conclusion

We present the NLU Showroom, a platform for demonstrating NLU models. The platform aims at people outside of the NLU community, who are intending to use NLU in their applications and products. In the current version, the NLU Showroom includes state-of-the art models for word and sentence similarity tasks, aspect and opinion mining and relation extraction, with more demos and models to follow. The platform builds on state-of-the-art open source components such as TensorFlow Serving, PyTorch and Docker.

In addition to releasing more demos we aim to link the showroom in the blog of the Competence Center for Machine Learning Rhine-Ruhr (ML2R). In the blog we will give detailed explanations about the models, tasks and data sets used to create the demos of the NLU Showroom and describe how these models can and have been used in projects and products.

References

1 AllenNLP demo. https://demo.allennlp.org. Accessed: 2021-02-02.

2 Docker. https://www.docker.com/. Accessed: 2021-02-02.

3 IBM Watson natural language understanding - text analysis. https://www.ibm.com/demos/live/natural-language-understanding/self-service/home. Accessed: 2021-02-02.

4 NLTK text processing demo. http://text-processing.com/demo/. Accessed: 2021-02-02.

5 Stanza online demo. http://stanza.run/. Accessed: 2021-02-02.

6 TorchServe. https://github.com/pytorch/serve. Accessed: 2021-02-02.

7 Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. URL: https://www.tensorflow.org/.

8 Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *CoRR*, abs/1607.04606, 2016. arXiv:1607.04606.

9 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi:10.18653/v1/N19-1423.

10 Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. AllenNLP: A deep semantic natural language processing platform. *arXiv*, 2017. arXiv:1803.07640.

11 Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden, 2010. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/S10-1006.

12 Matt Kusner, Y. Sun, N.I. Kolkin, and Kilian Weinberger. From word embeddings to document distances. *Proceedings of the 32nd International Conference on Machine Learning (ICML 2015)*, pages 957–966, January 2015.

13 Quoc V. Le and Tomas Mikolov. Distributed representations of sentences and documents. *CoRR*, abs/1405.4053, 2014. arXiv:1405.4053.

14 Omer Levy and Yoav Goldberg. Neural word embedding as implicit matrix factorization. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 2177–2185, Cambridge, MA, USA, 2014. MIT Press.

15 Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546, 2013. arXiv:1310.4546.

16 Christopher Olston, Noah Fiedel, Kiril Gorovoy, Jeremiah Harmsen, Li Lao, Fangwei Li, Vinu Rajashekhar, Sukriti Ramesh, and Jordan Soyke. Tensorflow-serving: Flexible, high-performance ML serving. *CoRR*, abs/1712.06139, 2017. arXiv:1712.06139.

17 Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

18 Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014. URL: http://www.aclweb.org/anthology/D14-1162.

19 Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland, 2014. Association for Computational Linguistics. doi:10.3115/v1/S14-2004.

20 Peng Qi, Timothy Dozat, Yuhao Zhang, and Christopher D. Manning. Universal dependency parsing from scratch. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 160–170, Brussels, Belgium, October 2018. Association for Computational Linguistics. URL: https://nlp.stanford.edu/pubs/qi2018universal.pdf.

21 Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2020. URL: https://nlp.stanford.edu/pubs/qi2020stanza.pdf.

22 Georg Rehm, Maria Berger, Ela Elsholz, Stefanie Hegele, Florian Kintzel, Katrin Marheinecke, Stelios Piperidis, Miltos Deligiannis, Dimitris Galanis, Katerina Gkirtzou, Penny Labropoulou, Kalina Bontcheva, David Jones, Ian Roberts, Jan Hajič, Jana Hamrlová, Lukáš Kačena, Khalid Choukri, Victoria Arranz, Andrejs Vasiļjevs, Orians Anvari, Andis Lagzdiņš, Jūlija Meļņika, Gerhard Backfried, Erinç Dikici, Miroslav Janosik, Katja Prinz, Christoph Prinz,

Severin Stampler, Dorothea Thomas-Aniola, José Manuel Gómez-Pérez, Andres Garcia Silva, Christian Berrío, Ulrich Germann, Steve Renals, and Ondrej Klejch. European language grid: An overview. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3366–3380, Marseille, France, 2020. European Language Resources Association. URL: https://www.aclweb.org/anthology/2020.lrec-1.413.

23 Martin Schiersch, Veselina Mironova, Maximilian Schmitt, Philippe Thomas, Aleksandra Gabryszak, and Leonhard Hennig. A german corpus for fine-grained named entity recognition and relation extraction of traffic and industry events. In *Proceedings of the 11th International Conference on Language Resources and Evaluation. International Conference on Language Resources and Evaluation (LREC-18), 11th, May 7-12, Miyazaki, Japan*. European Language Resources Association, 2018.

24 Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. Matching the blanks: Distributional similarity for relation learning. *CoRR*, abs/1906.03158, 2019. arXiv:1906.03158.

25 Chi Sun, Luyao Huang, and Xipeng Qiu. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. *CoRR*, abs/1903.09588, 2019. arXiv:1903.09588.

26 Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. Recursive neural conditional random fields for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 616–626, Austin, Texas, 2016. Association for Computational Linguistics. doi:10.18653/v1/D16-1059.

27 Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771, 2019. arXiv:1910.03771.

AAA4LLL – Acquisition, Annotation, Augmentation for Lively Language Learning

Bartholomäus Wloka
University of Vienna, Centre for Translation Studies, Vienna, Austria

Werner Winiwarter
University of Vienna, CSLEARN – Educational Technologies, Vienna, Austria

Abstract

In this paper we describe a method for enhancing the process of studying Japanese by a user-centered approach. This approach includes three parts: an innovative way of acquiring learning material from topic seeds, multifaceted sentence analysis to present sentence annotations, and the browser-integrated augmentation of perusing Wikipedia pages of special interest for the learner. This may result in new topic seeds to yield additional learning content, thus repeating the cycle.

2012 ACM Subject Classification Information systems → Browsers; Computing methodologies → Lexical semantics; Applied computing → E-learning

Keywords and phrases Web-based language learning, augmented browsing, natural language annotation, corpus alignment, Japanese computing, semantic representation

Digital Object Identifier 10.4230/OASIcs.LDK.2021.29

1 Introduction

As most of us know, learning a foreign language, unless it is done in early childhood, is a challenging task, demanding motivation, patience, and last but not least a good teacher or learning method. The endeavour becomes even more challenging when the language differs greatly from the ones we already know. We address this issue in this paper by proposing a self-directed, contextualized learning method for English speakers to learn Japanese.

Apart from the stark difference of the writing system, Japanese has a fairly unique style of grammar, heavily dependent on postpositions, the tendency to omit personal pronouns, and several registers of politeness, which are often expressed by entirely different verb forms. The Japanese writing style is not only different, but also uses a combination of the syllabary *kana*, which comes in two forms, *hiragana* and *katakana*, and a large collection of logographic characters, which are called *kanji*. Each of these pictograms has several possible readings and meanings and in most cases a complicated decomposition into smaller building blocks. Apart from that, their pronunciation, meaning, and grammatical function can be heavily modified by the embedding context. They can also be combined to form new compound expressions and terminology. Japanese children are taught kanji throughout their high school education with 80 to 200 kanji per school year [13]. This slow and gradual process of acquiring this writing system allows for a strong foundation and complex compound expressions and terms can be easily learned step by step.

Clearly, adult learners of Japanese do not have this luxury and the complex characters, grammar, and spoken Japanese have to be learned at the same time. To put things in perspective, there are far more characters in everyday use and kanji are learned throughout the entire adult life in Japan. Standard dictionaries include more than 10,000 characters and over 50,000 words built from these characters. This means that learning methods need to streamline the process as much as possible just to assist the learner in catching up with this life long learning process of a native Japanese speaker.

It is clear, that a learning environment needs to be interesting and engaging in order to increase the students' motivation. We find that this is best achieved by granting the learners the most possible freedom in selecting the material, while leading them towards understanding by offering the best possible decomposition of difficult concepts and their explanation. After presenting a possible translation to the learners, we deconstruct the Japanese sentence and enhance it with lexical, syntactic, conceptual, and relational annotation. The additional information is presented in a visually appealing way using colors and images to improve the overview of the structure. The learners can, of course, adjust the level of detail to their current language skill. We call these two parts *Acquisition* and *Augmentation of Learning Contexts*.

We first discuss the relevant related work in Sect. 2. We then describe the technical details of the implemented proof-of-concept framework in Sect. 3. In Sect. 4 we explain the automatic preparation of the learning material starting from seed topics chosen by the learner. In Sect. 5 we provide a detailed intuitive example of the augmented display presented to the learner. We summarize in Sect. 6 with a discussion, and our plans on how to extend and evaluate our approach in an in-class setting.

2 Related Work

The idea we build on in this paper is that motivation, the ability to choose the content, a keen interest in the subject matter, and a multimodal and multilayered environment are key to success when learning a new language. This is discussed extensively in [15]. A different approach to the same idea is shown in the incidental learning technique in [17], where the students are presented with information while browsing on-the-fly and in an unintrusive way. While browsing content of interest in the language they learn, like articles about their hobbies or daily news, the students are supported with facts about the text; a very elegant method which, however, clearly requires a relatively high level of skill in the foreign language, hence is reserved for advanced learners.

We extend on this concept in this paper on several levels, including the opportunity to use it even earlier in the language learning process. We do this by letting the learner decide the context, while still maintaining a feasible didactic structure and sensible levels of difficulty of the content. Research such as [14] shows the effectiveness of this approach. The interviews carried out in [6] further support these findings from the subjective point of view of the learners.

The component that makes this possible is the pre-selection of context, for which we use a bilingual alignment method of Wikipedia content, based on a metric obtained by matching of lexical units. This method and a discussion about how well Wikipedia is suited as a source of parallel content for the Japanese-English language pair can be found in [18].

Research in alignment and harvesting of bilingual material has been very active in the last decades, especially due to the importance of the application of such data in machine translation and other language technologies requiring training data. We would like to mention the largest of these approaches including the Japanese-English language pair, *JParaCrawl* [10], resulting in a collection of 10 million Japanese-English sentence pairs. The *WikiMatrix* project produced a multilingual collection with a large amount of parallel language data in 85 languages. This was done by using LASER sentence embeddings [1]. With such a volume of data, the quality varies greatly between the language pairs and is comparatively low for Japanese-English. These two large scale approaches are representative of the current paradigm of multilingual data collection, namely the brute-force black box approach. The

quality and huge computational requirements problems – an issue raised well in [3] – aside, the lack of transparency or a quality score makes it difficult to trace a path to the result and to judge the fitness of aligned data for language learning.

For the lexical and syntactic annotation of the sentences we use the de facto standard *universal dependencies* [11], whereas we rely on the *abstract meaning representation* (AMR) [2] for the semantic representation. Recently, there has been a renewed interest on different approaches for meaning representation (for a recent shared task see [12]). We have chosen AMR over other competing approaches because we judged it most suitable for language learning purposes.

3 System Architecture

We have implemented a language learning environment in which the users can study Japanese Wikipedia pages based on topics that really interest them. The learners select one or several topics as seeds. The following architecture turns them into a topic-specific collection of parallel sentences.

The architecture is divided into three stages, and each stage is further divided into modules. The modular approach allows for flexibility and ease of maintenance. Figure 1 gives an overview of this architecture.

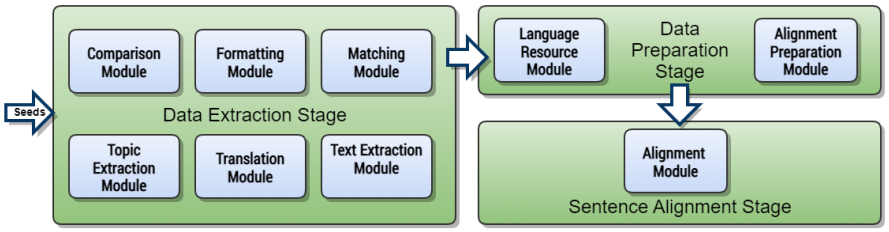


Figure 1 Stages with their corresponding modules.

The flow of the data through this pipeline of stages and modules starts at the top with the *Data Extraction Stage*. This stage processes the seed input and extracts text data from Wikipedia accordingly. This is done by processing each English Wikipedia page of a seed topic and finding all links to further pages within this article. This process is repeated for the according Japanese pages. This is taken as the first measure of similarity between the contents. A discussion and preliminary results for the question of how much of the content between Japanese and English articles is comparable can be found in [18]. We use a threshold value to adjust the degree of similarity vs. the volume of candidate data.

Since we traverse the pages recursively and define each link as the starting point for the next iteration we obtain text that is in some way related to the initial seed. Naturally, the semantic distance increases with the number of links from the initial topic, which can also be adjusted as needed. Once we have collected topics, which we deem as good candidates, we extract the text from the Wikipedia pages.

In the *Data Preparation Stage* we prepare the text for further processing, i.e., the alignment. We segment the English text on a sentence and word level, lemmatize it, and determine the part-of-speech. We use dictionary resources from *EDRDG*[1]. We segment and annotate the Japanese text with *MeCab* [8], which we later also use for our *lexical annotation*.

[1] http://www.edrdg.org/

Once the data is prepared, we align the bilingual input in the *Sentence Alignment Stage*. With the assumption that at least a good portion of the data has translation equivalents – depending on the above-mentioned threshold value – we traverse the entire pre-processed data set for matches. Selected lexical units in the Japanese sentence are examined for potential alignment indicators. An example of this process is shown in Fig. 2.

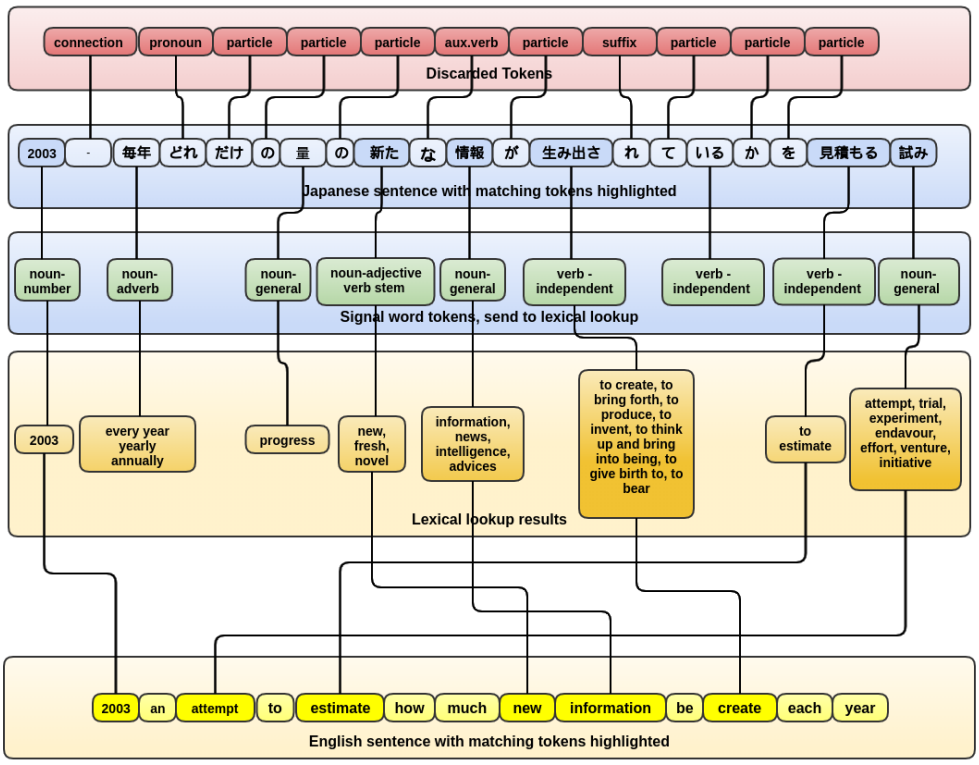


Figure 2 Alignment example.

On top of the schematic depiction of this alignment process we see the tokenized Japanese sentence. Above the sentence is a row with discarded PoS tokens, below are the tokens taken into account for alignment. The reason we discard some of the tokens is that they contribute less to the alignment process. A detailed discussion can be found in [19].

The remaining tokens are being looked up in the lexical resources, i.e. bilingual dictionaries. It is important to mention that we retrieve each possible lexical equivalent of the words, as shown in the line with the lexical lookup results in Fig. 2.

Translations of sentences often vary in style and several differently sounding sentences might convey the same content. We consider these variations with our alignment method that takes into account all possible synonyms. This helps us to identify several – often stylistically varying – candidates, which is particularly interesting in a learning context.

In the process of matching the individual parts we compute an alignment score based on the number of matches normalized by the sentence length. We use this score, which indicates the alignment quality, hence the lexical similarity between the Japanese and the English sentence, to sort by alignment confidence. The output, i.e. the set of best alignment candidates, is the input to the annotated presentation.

For the purpose of presenting the annotated view to the learners we have designed a Web-based client-server architecture using *augmented browsing* technology to enhance the Web documents with event handlers at the client to retrieve annotations from the learning server and display the information in a comprehensible way. This is realized with the *WebExtensions API cross-browser technology*[2] and the *jQuery*[3] and *jQueryUI*[4] libraries.

The language learning server is implemented in *SWI-Prolog*[5]. It is not only an excellent choice for natural language processing tasks but also offers a scalable Web server solution as well as libraries for the efficient handling of huge XML and RDF files. The annotation data is transferred via XMLHttpRequests in JSON and assembled at the client in popup divs.

The only external software that we use is the Japanese dependency parser *CaboCha* [7]. We take its output as starting point for analyzing and annotating a sentence in four steps: the lexical, syntactic, conceptual, and relational level. We use several lexicosemantic resources: the dictionary files from EDRDG, *WordNet* [9], and datasets from *DBpedia*[6].

For *lexical annotation*, we take the output of CaboCha, which includes the above-mentioned part-of-speech and morphological analyzer MeCab. The latter uses a fine-grained hierarchical tagset with up to four levels and additional conjugation types and forms. We map this tagset to *universal POS tags*[7]. Since MeCab follows a rather extreme segmentation strategy, which we found quite unsuitable for educational purposes, we use our lexical resources to merge adjacent tokens to achieve a more compact and comprehensible presentation. For the *kanji cards* in our display, we include *ideographic description sequence* data[8] from the *CHISE* project[9]. We also show images as visual clues for kanji, which were hand-collected from Wikipedia pages.

CaboCha transforms a sentence into a sequence of *segments*, which are linked through *dependency patterns*. As CaboCha does not output any syntactic relation names, we had to add the appropriate *universal syntactic relation* names for displaying the *syntactic annotation*. We also had to arrange the relations vertically into several rows to offer an appealing visual representation.

The *conceptual annotation* is based on XML frame files from the *OntoNotes* project available from *LDC*[10], and *AMR resource lists*[11]. We extend the AMR approach by mapping words also to Wikipedia pages through DBpedia disambiguation links and to WordNet synsets, whenever we cannot find a suitable frame. We also display short abstracts and thumbnails for Wikipedia pages, again retrieved from DBpedia. For disambiguation, we rely on contextual and distributional data from the current sentence and its English equivalent, the Wikipedia page, and the collected topic-specific corpus.

As a final step, we add *roles* to the display, using again data from the OntoNotes frame files to offer a *relational annotation* to round off our augmented view of the linguistic and semantic properties of a Japanese sentence. In Sect. 5, we go through a detailed example, which illustrates the individual annotation levels as perceived from the perspective of the language student.

---

[2] `https://developer.mozilla.org/en-US/docs/Mozilla/Add-ons/WebExtensions`
[3] `https://jquery.com`
[4] `https://jqueryui.com`
[5] `https://www.swi-prolog.org/`
[6] `https://wiki.dbpedia.org/`
[7] `https://universaldependencies.org/guidelines.html`
[8] `https://github.com/cjkvi/cjkvi-ids`
[9] `https://www.chise.org/`
[10] `https://www.ldc.upenn.edu/`
[11] `https://amr.isi.edu/download.html`

The use of SWI-Prolog for implementation has the big advantage that all the data is stored and accessed in a declarative way as Prolog *fact files*, which can be easily customized and reconsulted dynamically. We also developed several visual interfaces and editors for expert users in previous research efforts (see [16, 17]).

4 Acquisition of Learning Contexts

The initial step in our learning environment is the selection and acquisition of the learning material. Since we firmly believe that the best way to learn is to examine interesting content while having the difficulty of the material custom shaped to the language level, rather than reading boring texts that clearly only aim at conveying vocabulary or a grammatical concept, we let the user choose their context freely. The initial step is to find engaging content on English Wikipedia. A topic or several related topics of interest then become the seed(s) towards a collection of example sentences to start the learning journey, while discovering the desired information about a certain topic on Wikipedia.

The sentences are selected according to the description in Sect. 3. Thanks to the efficiency of the alignment algorithm, the learners can extract new information about any topic within minutes or a few hours, depending on the size of the dataset. Naturally, the exact time of the sentence alignment depends on the desired size of the example sentence corpus. Table 1 shows an example of collecting 805 example sentences for one seed topic.

Table 1 Runtime example for a small dataset.

Module	Time	Output
Data Extraction Stage		
Topic Extraction	30m36s	en: 801087 articles, ja: 56736 articles
Text Extraction	18.1s	en: 2037 lines, ja: 2072 lines en: 85,804 tokens, ja: 75,393 tokens
Data Preparation Stage		
Alignment Preparation	59m45s	en: 3510 sentences, ja: 805 sentences
Sentence Alignment Stage		
Alignment	24m50s	805 aligned sentences

It is important to mention that the runtime efficiency increases with the number of alignments run on the learner's computer, since frequently occurring topic equivalents are stored locally and do not have to be looked up repeatedly in the Wikipedia database. This and other runtime tweaks are described in detail in [19]. After the output is generated, the learners can select which and how many of the best scoring sentences they want to examine. With the sentences they have chosen, the users then continue their learning experience with the Augmentation part of our language learning solution.

5 Augmentation of Learning Contexts

In this section we present the augmented information displayed to the user while working with Japanese Wikipedia pages as study material. We discuss each annotation level in a cleanly separated subsection. Throughout this section, we use one sentence as running example, taken from the Japanese Wikipedia page on *thrust* [12]. Figure 3 shows the complete pop-up div, which appears when the student clicks on the sentence in the Web page.

[12] https://ja.wikipedia.org/wiki/%E6%8E%A8%E5%8A%9B

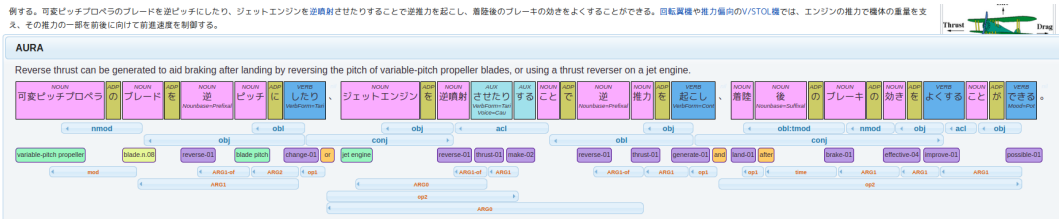


Figure 3 Example of augmentation.

Unfortunately, the sentence is rather long and complicated, which is fairly typical for sentences found on Japanese Wikipedia pages. Luckily, our acquisition step (see Sect. 4) equips us with a reasonably good translation from the corresponding English Wikipedia page to significantly facilitate the challenging task for the language learner to make any sense of this text. To offer sufficient resolution for readability, we divide the presentation of the sentence into three parts in the following subsections. Since Japanese grammar is exclusively head-final and strongly left-branching with abundant use of postpositional particles, we consequently go through the sentence from right to left.

5.1 Lexical Annotation

We annotate the sentence using *universal POS tags* for the individual tokens, which are also visually emphasized through different colors. In our example sentence, these are: *blue* for verbs, *light blue* for auxiliaries, *pink* for nouns, and *olive* for adpositions.

Whenever necessary, we add *universal features*. In some cases, we also decided to use *language-specific features* and *values*. All these choices can be easily adjusted by expert users to accommodate their personal preferences.

The user can click on each token to open a popup div with further information. This includes the English glosses retrieved from our lexical resources and *kanji cards* for the kanji that are part of the word. The kanji cards include the *radical* number, *on'yomi* readings (in uppercase), *kun'yomi* readings, and English glosses. Radicals are 214 special kanji that are used as components of other kanji. One of them is always singled out as the radical of a kanji to look up the character in a kanji dictionary. On'yomi readings descend from approximations of original Chinese pronunciations whereas kun'yomi readings are based on pronunciations of native words approximating the meaning of the character when it was introduced. Finally, we display an image to offer a visual clue for memorizing the kanji. The *ideographic description sequence* defines the spatial structure of the kanji based on simpler components, the radical of the kanji is highlighted in *red* in this sequence, other radical components in *orange*.

In Fig. 4, we show the information for the noun 着陸 (`chakuriku`). As can be seen, the correct contextual pronunciations of the kanji are indicated in pink in the lists of possible readings.

The right part of the sentence contains the following lexical tokens for content words:

- the verb できる (`dekiru`), which is actually the *potential* form of the verb する (`suru`) "to do", therefore, meaning "to be able to do";
- the noun こと (`koto`) "thing", which just nominalizes the preceding clause;
- the verb よくする (`yokusuru`) "to improve";
- the noun 効き (`kiki`) "effectiveness", derived from the *continuative* form of the verb `kiku`;

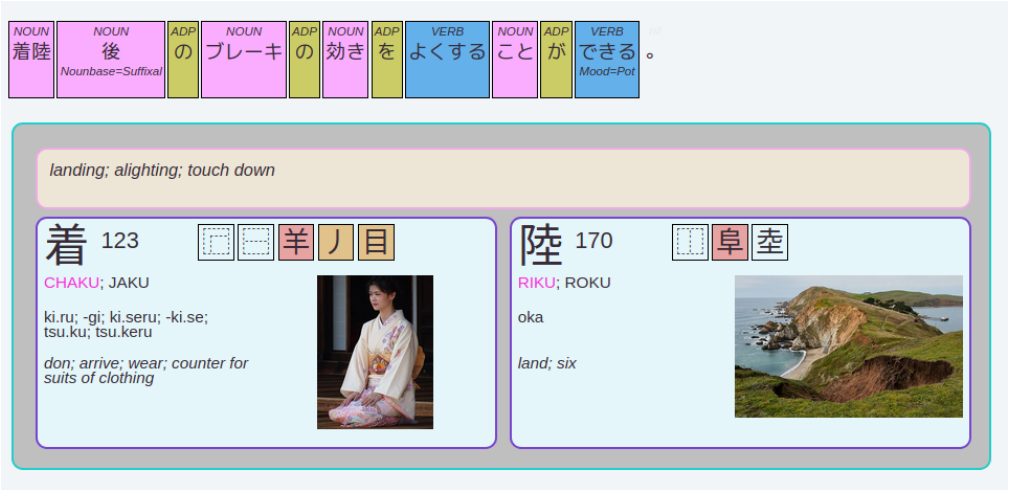


Figure 4 Example of lexical analysis – right part.

- the loanword ブレーキ (`burēki`) “brake”;
- the suffix 後 (`go`) “after”, which is used like a *temporal* adposition; and
- the noun 着陸 (`chakuriku`) shown in the popup div.

Since adpositions and auxiliaries mainly serve as syntactic function words, we discuss them later in Sect. 5.2. Figure 5 shows the middle part of the sentence with the following lexical units:

- the continuative form 起こし (`okashi`) of the verb 起こす (`okasu`), which here means “to generate”, the continative form is used like the conjunction “and” to loosely connect the two clauses;
- the noun 推力 (`suiryoku`) “thrust”;
- the prefix 逆 (`gyaku`) “reverse”, shown in the popup div;
- the noun 逆噴射 (`gyakufunsha`) “reverse thrust”; and
- the loanword ジェットエンジン (`jettoenjin`) “jet engine”.

Finally, the left part of the sentence, as shown in Fig. 6, consists of the following lexical units:

- the *tari*-form したり (`shitari`) of the verb `suru`, which is used like the conjunction “or ” to connect several exemplars with the pattern `-tari ...-tari suru`;
- the loanword ピッチ (`pitchi`), meaning here “blade pitch” or “angle”;
- the loanword ブレード (`burēdo`) “blade”; and
- the compound noun (part loanword) 可変ピッチプロペラ (`kahenpitchipuropera`) “variable pitch propeller”, for which the details are shown in the popup div.

5.2 Syntactic Annotation

For the syntactic annotation, we add *universal syntactic relations* to the display. However, to improve the comprehensiveness of the visual representation, we omit obvious relations between adjacent tokens. As mentioned in Sect. 3, we use the output of the Japanese dependency parser *CaboCha* for this purpose, but enhance it with syntactic relation names. Figure 7 shows the dependencies for the right part of our example sentence:

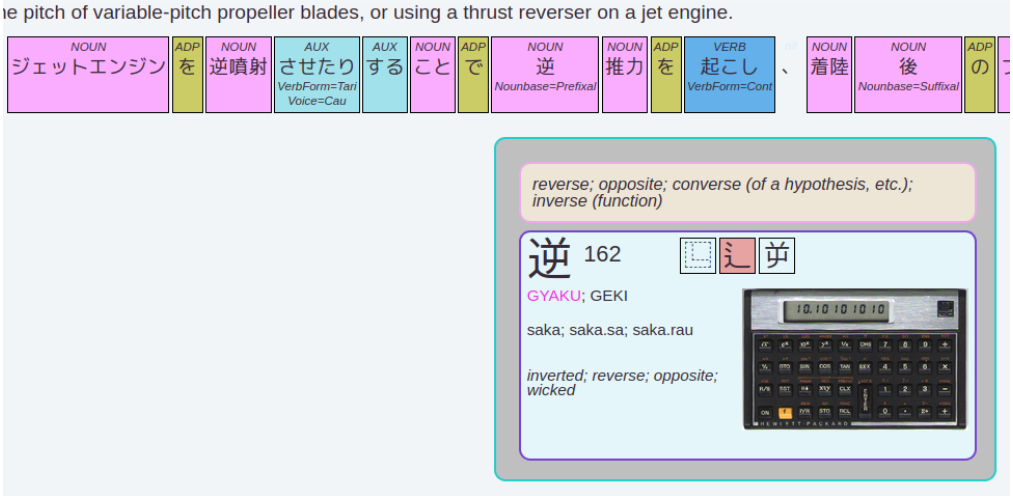


Figure 5 Example of lexical analysis – middle part.

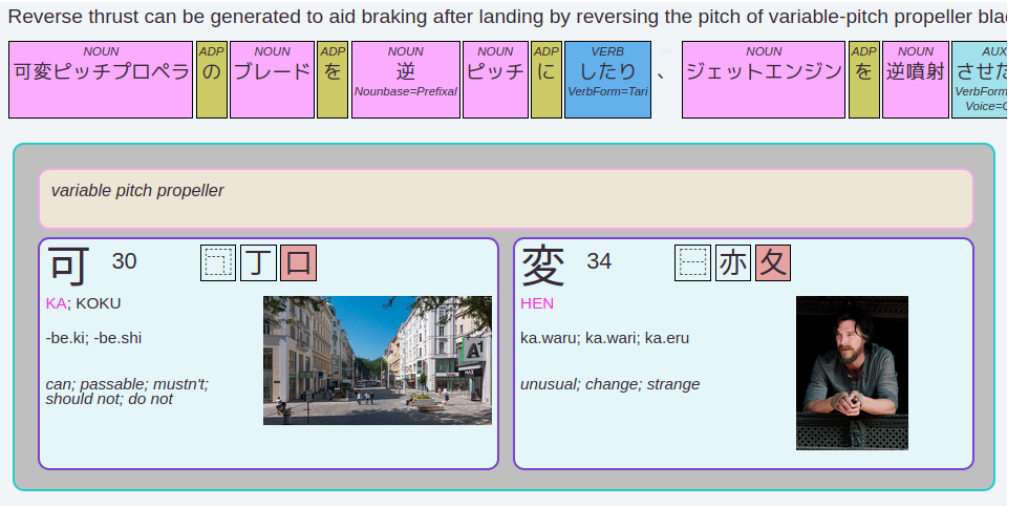


Figure 6 Example of lexical analysis – left part.

Figure 7 Example of syntactic annotation – right part.

- the *object* **obj** relation between `dekiru` and `koto`, indicating the "thing" that "can be done", the postposition カ゛ (`ga`) usually marks the subject, however, in this context the direct object of the verb;
- the **acl** relation between `koto` and `yokusuru`: as mentioned before, this *adnominal clause* relation nominalizes the preceding clause, resulting in "improvement", however, in combination with `koto ga dekiru` this effect is somehow canceled out as it just means "it is possible to improve";

- the **obj** relation between `yokusuru` and `kiki` tells us that the "effectiveness can be improved", here we can see the usual direct object marker を (`o`);
- the *nominal modifier* **nmod** relation between `kiki` and `burēki` with the corresponding adposition の (`no`) means that the former is an attribute or genitive complement, i.e. "the effectiveness of the brakes" or "the effectiveness of braking";
- because of the special meaning "after" of the suffix `go`, we have a *temporal modifier* **obl:tmod** relation between `burēki` and `chakuriku`: "braking after landing";
- finally, there is a *conjunct* **conj** relation between this clause and the preceding clause shown in Fig. 8, due to the continuative form `okashi`, as mentioned before.

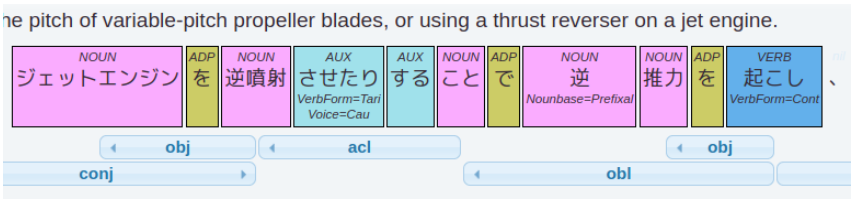


Figure 8 Example of syntactic annotation – middle part.

The middle part of the sentence, as shown in Fig. 8, contains the following relations:

- `suiryoku` is the direct object of `okashi`, i.e. we "generate thrust";
- there is an *oblique nominal* **obl** relation between `okashi` and `koto`, the extremely polysemous adposition で (`de`) indicates here the means by which we generate the thrust;
- the two auxiliaries させたり (`sasetari`) and する (`suru`) combine with the preceding noun `gyakufunsha` and verbalize it so we end up with something like "reverse thrusting", again this is undone by the **acl** relation with the noun `koto`,
- the **obj** relation to "jet engine" shows that we "reverse thrust the jet engine", actually, here the *causative* form of the verb is used, so it literally means "make reverse thrust the jet engine";
- ultimately, we have again a **conj** relation to the left part of the sentence, thanks to the already mentioned `-tari ...-tari suru` construction.

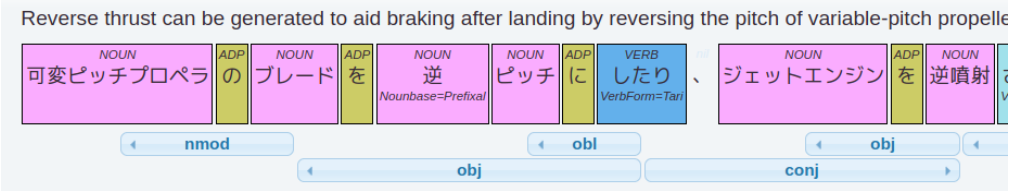


Figure 9 Example of syntactic annotation – left part.

The final left part of the sentence in Fig. 9 only contains three additional relations:

- there is an **obl** relation between `shitari` and "pitch", together with the again very polysemous adposition に (`ni`) this indicates here that we change something to a new state, i.e. "reversed pitch";
- what we change is expressed by the **obj** relation, namely the "blade"; and
- to clarify matters through an **nmod** relation, it is the "blade" of a "variable pitch propeller".

5.3 Conceptual Annotation

At the third level of annotation we map content words to concepts within the semantic representation framework AMR. In AMR we can distinguish between dedicated *AMR frames* like the one shown in Fig. 10 (*orange*) and *OntoNotes frames* as displayed in Fig. 11 (*purple*). As usual, the popup divs can be inspected by just clicking on the concept names. The frames provide a definition and several *roles* (see Sect. 5.4). For the right part of the sentence, we can see the following mappings:

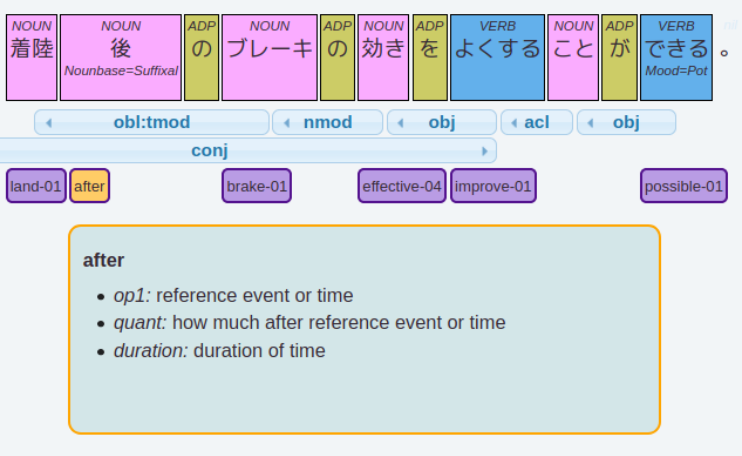


Figure 10 Example of conceptual annotation – right part.

- `dekiru` ⇒ `possible-01`: "likely or able to be/occur",
- `yokusuru` ⇒ `improve-01`: "make better",
- `kiki` ⇒ `effective-04`: "cause an effect, successful in creating a desired effect",
- `brēki` ⇒ `brake-01`: "slow a car via brakes",
- `go` ⇒ `after`,
- `chakuriku` ⇒ `land-01`: "bring to land, from water or air".

We use the glosses from the lexical annotation to retrieve possible frames, and contextual and distributional data to disambiguate among likely candidates (see Sect. 3). As can be seen in Fig. 11, conjunctions are mapped in AMR to special `and` and `or` frames, in addition, there are the following mappings to OntoNotes frames for the middle part:

- `okoshi` ⇒ `generate-01`: "create",
- `suiryoku` ⇒ `thrust-01`: "to push quickly and forcibly",
- `gyaku` ⇒ `reverse-01`: "turn around, change direction",
- `gyakufunsha sasetari suru` ⇒ `reverse-01`, `thrust-01`, and `make-02`: "cause (to be)".

The last entry is an example of assigning several concepts to one position in the sentence: the first two concepts correspond to the noun `gyakufunsha`, the last one is represented by the auxiliaries `sasetari suru`, however, since auxiliaries are not annotated as dependencies by CaboCha, we also map the third concept to the noun. In Fig. 11 there is also one mapping of a word to the corresponding Wikipedia page: `jettoenjin` ⇒ "jet engine" (*green*). Whenever we cannot map a word to a frame, we try to find a Wikipedia page representing the concept. If a user clicks on such a concept name, we display the short abstract and thumbnail retrieved via DBpedia (see Fig. 12). We use mainly the *DBpedia disambiguation links* data to identify

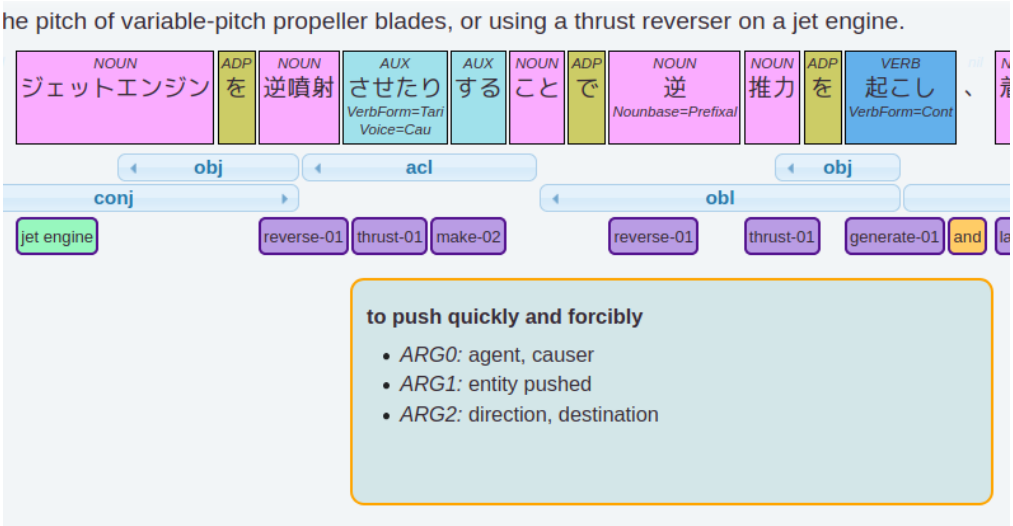


Figure 11 Example of conceptual annotation – middle part.

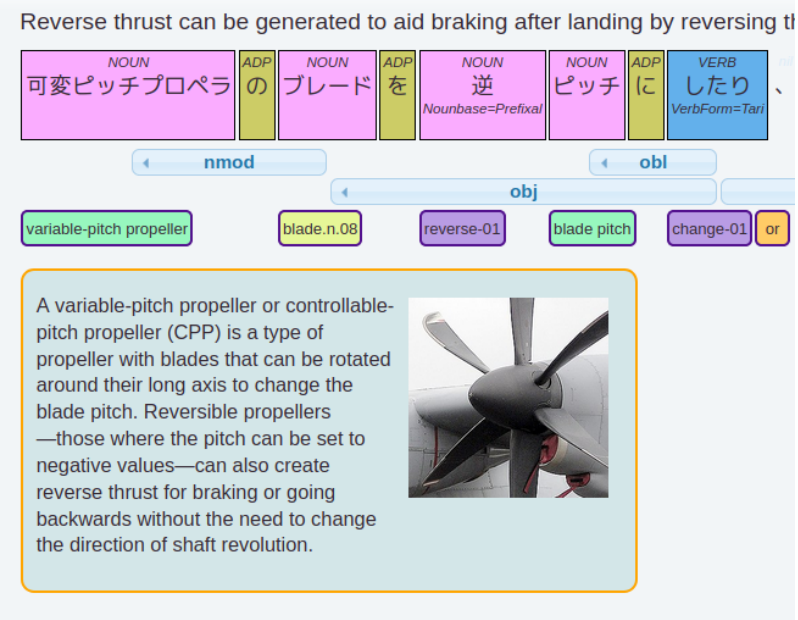


Figure 12 Example of conceptual annotation – left part.

ambiguous words and select the Wikipedia page representing the correct word sense. Finally, if there is no existing Wikipedia page, we use *WordNet* as backup, again retrieving the correct *synset* (*yellow*) using word sense disambiguation based on contextual data and relational information derived from WordNet. If we click on a WordNet concept, the synset definition is displayed. This results in the following mappings for the left part of the sentence:

- `shitari` ⇒ `change-01`: "transform",
- `pitchi` ⇒ `blade pitch`,
- `burēdo` ⇒ `blade.n.08`: "flat surface that rotates and pushes through air or water",
- `kahenpitchipuropera` ⇒ `variable-pitch propeller`.

As can be seen, we can successfully narrow down the senses of the polysemous word "pitch" by following the disambiguation link to the correct Wikipedia page on `blade pitch`. In the case of "blade", this is not possible, because there is only a Wikipedia page for the sense "sharp cutting part, for instance of a weapon or tool".

5.4 Relational Annotation

With the relational annotation level, we complete the picture by adding semantic roles to the display to offer a semantic representation of the meaning of the sentence within the AMR framework. As can be seen in Fig. 13, we use a visual representation similar to that of universal dependencies in the syntactic annotation.

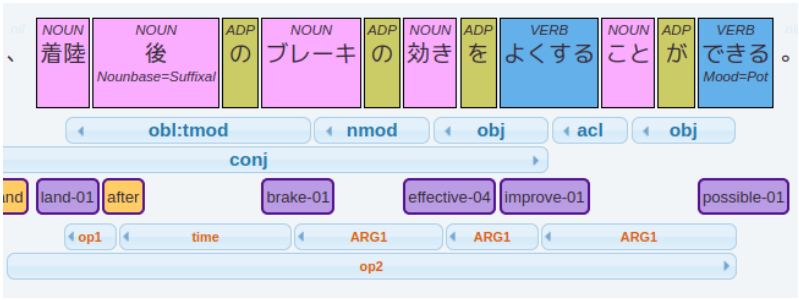


Figure 13 Example of relational annotation – right part.

Whenever possible, we use *core roles*, defined in the OntoNotes frames (`ARG0, ARG1, ...`). In addition, AMR offers an inventory of *non-core roles*, e.g. `time` in Fig. 13 indicates the time when the braking occurs. The roles `op1, op2, ...` are special roles only used in AMR frames. Therefore, we have the following roles for the right part of the sentence:

- `possible-01` $\xrightarrow{\text{ARG1}}$ `improve-01`: `improve-01` is the "thing that is possible",
- `improve-01` $\xrightarrow{\text{ARG1}}$ `effective-04`: `effective-04` is the "thing improving",
- `effective-04` $\xrightarrow{\text{ARG1}}$ `brake-01`: `brake-01` is the "domain in which arg0 (cause) is effective; outcome effected".

We only indicate roles for which there is explicit evidence in the Japanese sentence. Since Japanese omits many details that are usually expressed in other languages at least through anaphora (a phenomenon also known as *zero anaphora* [5]), it is not often necessary to use the variable mechanism of AMR to refer to antecedents. The middle part of the sentence (see Fig. 14) contains the following roles, it also shows the use of *inverse roles* to *re-focus* the AMR representation:

- `generate-01` $\xrightarrow{\text{ARG1}}$ `thrust-01`: `thrust-01` is the "thing created",
- `thrust-01` $\xrightarrow{\text{ARG0}}$ `or`: the whole `-tari ...-tari suru` construct is the "agent, causer",
- `thrust-01` $\xrightarrow{\text{ARG1-of}}$ `thrust-01`: this is an inverse role indicating that `thrust-01` is the "thing turning around",
- `make-02` $\xrightarrow{\text{ARG1}}$ `thrust-01`: `thrust-01` is the "impelled action/ predication",
- `thrust-01` $\xrightarrow{\text{ARG0}}$ `jet engine`: in this case, `jet engine` is the "agent, causer".

Finally, Fig. 15 displays the semantic roles for the remaining right part of the sentence:

- `change-01` $\xrightarrow{\text{ARG1}}$ `blade.n.08`: `blade.n.08` is the "thing changing",

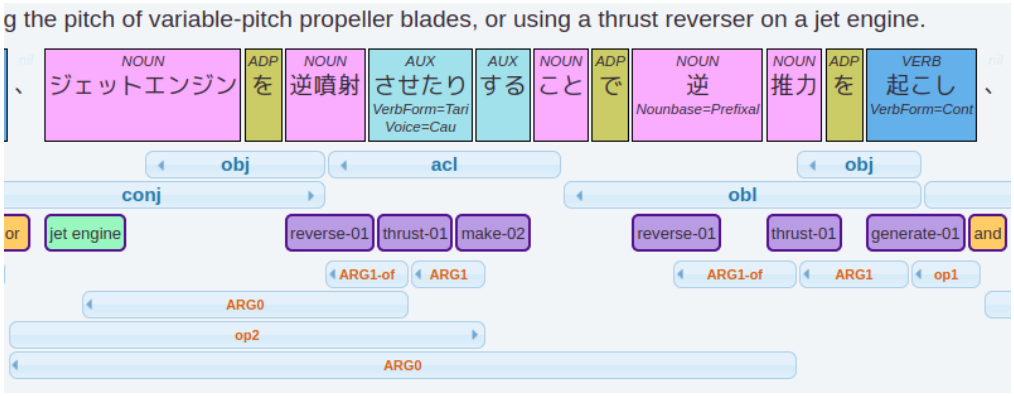


Figure 14 Example of relational annotation – middle part.

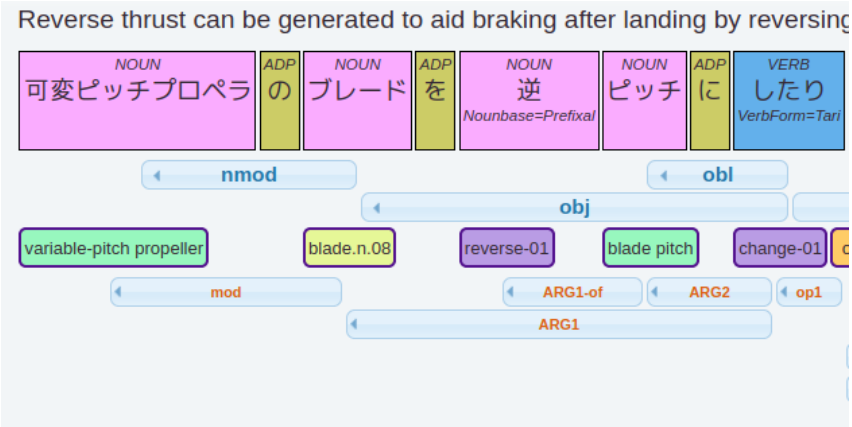


Figure 15 Example of relational annotation – left part.

- `change-01` $\xrightarrow{\text{ARG2}}$ `blade pitch`: `blade pitch` is the "end state",
- `blade.n.08` $\xrightarrow{\text{mod}}$ `variable-pitch propeller`: this non-core role tells us that the latter is a *modifier* of `blade.n.08`.

6 Conclusion

We have presented a Lively Language Learning solution that enables the student to explore customized material in a dynamic way through Acquisition, Annotation, and Augmentation (AAA4LLL). We have described how the users can choose their learning context by selecting seed topics, finding appropriate translations of interesting sentences from Wikipedia with the help of a transparent and traceable alignment technique, and inspecting these sentences by studying the individual parts, enriched with lexical, syntactic, conceptual, and relational annotation. The learning process can then be repeated by selecting new or additional topics.

As future work we will evaluate our learning solution in a classroom setting, for which we will involve graduate level language students. We will assess both system performance and learning outcomes by additionally employing novel evaluation approaches, such as learner centered development as described in [4]. Finally, we are planning to release our learning environment as open software together with instructive demos and an extensive documentation of the annotation formats.

References

1 Mikel Artetxe and Holger Schwenk. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions ACL*, 7:597–610, 2019. doi:10.1162/tacl_a_00288/43523.

2 Laura Banarescu et al. Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186. ACL, 2013. URL: https://www.aclweb.org/anthology/W13-2322/.

3 Marta Bañón et al. ParaCrawl: Web-scale acquisition of parallel corpora. In *Proceedings of the 58th Annual Meeting of the ACL*, pages 4555–4567. ACL, 2020. doi:10.18653/v1/2020.acl-main.417.

4 Hendrik Heuer and Daniel Buschek. Methods for the design and evaluation of HCI+NLP systems. *arXiv*, 2102.13461 [cs.CL], 2021. arXiv:2102.13461.

5 Ryu Iida, Kentaro Torisawa, Jong-Hoon Oh, Canasai Kruengkrai, and Julien Kloetzer. Intra-sentential subject zero anaphora resolution using multi-column convolutional neural network. In *Proceedings EMNLP 2016*, pages 1244–1254. ACL, 2016. doi:10.18653/v1/d16-1132.

6 Maki Kubota. Post study abroad investigation of kanji knowledge in Japanese as a second language learners. *System*, 69:143–152, 2017. doi:10.1016/j.system.2017.07.006.

7 Taku Kudo and Yuji Matsumoto. Fast methods for kernel-based text analysis. In *Proceedings of the 41st Annual Meeting of the ACL*, pages 24–31. ACL, 2003. doi:10.3115/1075096.1075100.

8 Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying conditional random fields to Japanese morphological analysis. In *Proceedings EMNLP 2004, ACL 2004*, pages 230–237. ACL, 2004. URL: https://www.aclweb.org/anthology/W04-3230/.

9 George A. Miller. WordNet: A lexical database for English. *Commun. ACM*, 38(11):39–41, 1995. doi:10.1145/219717.219748.

10 Makoto Morishita, Jun Suzuki, and Masaaki Nagata. JParaCrawl: A large scale web-based English-Japanese parallel corpus. In *Proceedings of LREC 2020*, pages 3603–3609. ELRA, 2020.

11 Joakim Nivre et al. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of LREC 2020*, pages 4034–4043. European Language Resources Association, 2020. URL: https://www.aclweb.org/anthology/2020.lrec-1.497/.

12 Stephan Oepen et al. MRP 2020: The second shared task on cross-framework and cross-lingual meaning representation parsing. In *Proceedings of the CoNLL 2020 Shared Task: Cross-Framework Meaning Representation Parsing*, pages 1–22. ACL, 2020. doi:10.18653/v1/2020.conll-shared.1.

13 Simon Paxton. Kanji matters in a multilingual Japan. *The Journal of Rikkyo University Language Center*, 42:29–41, 2019.

14 Harald Wahl and Werner Winiwarter. A technological overview of an intelligent integrated computer-assisted language learning (iiCALL) environment. In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications*. AACE, 2011.

15 Werner Winiwarter. Mastering Japanese through augmented browsing. In *Proceedings of iiWAS 2013*, iiWAS '13, pages 179–188. ACM, 2013.

16 Werner Winiwarter. JAMRED: a Japanese Abstract Meaning Representation EDitor. In *Proceedings of iiWAS 2015*, pages 11:1–11:5. ACM, 2015. doi:10.1145/2837185.2837246.

17 Werner Winiwarter. JILL: Japanese Incidental Language Learning. In *Proceedings of iiWAS 2015*, pages 9:1–9:9. ACM, 2015. doi:10.1145/2837185.2837191.

18 Bartholomäus Wloka. Identifying bilingual topics in Wikipedia for efficient parallel corpus extraction and building domain-specific glossaries for the Japanese-English language pair. In *Proceedings of LREC 2018*. ELRA, 2018.

19 Bartholomäus Wloka. *Automated Creation of Domain-Specific Bilingual Corpora for Machine Translation, focusing on Dissimilar Language Pairs*. PhD thesis, University of Vienna, 2020.

Improving Intent Detection Accuracy Through Token Level Labeling

Michał Lew
SentiOne Research, Gdańsk, Poland

Aleksander Obuchowski
SentiOne Research, Gdańsk, Poland
Gdańsk University of Technology, Poland

Monika Kutyła
SentiOne Research, Gdańsk, Poland

Abstract

Intent detection is traditionally modeled as a sequence classification task where the role of the models is to map the users' utterances to their class. In this paper, however, we show that the classification accuracy can be improved with the use of token level intent annotations and introducing new annotation guidelines for labeling sentences in the intent detection task. What is more, we introduce a method for training the network to predict joint sentence level and token level annotations. We also test the effects of different annotation schemes (BIO, binary, sentence intent) on the model's accuracy.

2012 ACM Subject Classification Computing methodologies → Natural language processing

Keywords and phrases Intent Detection, Annotation, NLP, Chatbots

Digital Object Identifier 10.4230/OASIcs.LDK.2021.30

1 Introduction

Intent detection is a part of the Natural Language Understanding (NLU) component used in intelligent dialog systems and chat-bots. It is responsible for capturing the intention behind users' utterances based on semantics. Intent detection has been traditionally modeled as a sentence classification task where a whole utterance is mapped to its label. This approach, however, imposes certain limitations, such as problematic representations of sentences with multiple intentions or multi-sentence utterances, in which users want the agent to perform several tasks at once.

In this paper, we propose a token-level sentence annotation method capable of improving the intent classification accuracy. This approach is motivated by the fact that certain words contain stronger semantic properties with respect to the user's intention. Identifying and labeling these words in the annotation process helps to provide additional knowledge to the classifier. Token level information sends additional signals to the neural network, helping with error propagation and generalization of the models. Token intents have also previously been shown to help assign constrains to multi-intent sentences and improve capturing dependencies between those intents [4]. Additionally, we present a method of joint training of the token-level intent labels and sentence level labels in the multi-task learning fashion. Using this type of training helps prevent the loss of information across the network resulting in its final accuracy.

In our experiments, we present the improvement in model's preference by using token-level intent information instead of utterance-level labels. Models trained in this fashion are able to achieve better results in the intent detection task. We also compare different annotation schemes (BIO vs binary), as well as different token level information generation methods.

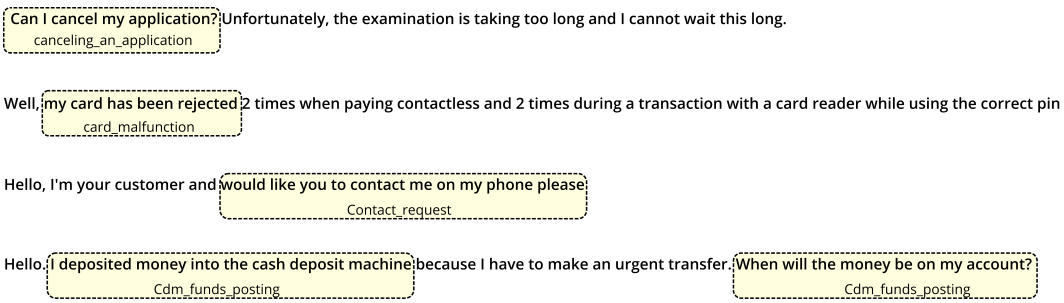


Figure 1 Examples user utterances and their annotations focusing on specific parts of the sentences indicating the intent.

2 Related Work

Intent detection has been modeled as a classification task where every sentence is assigned a label corresponding to its intent. Several neural network architectures have been proposed for intent recognition. These solutions were mainly based on BiLSTM [6, 8] networks. Recently state-of-the art solutions use capsule neural networks as well [15, 16, 10]. In our experiments we use architecture based on the widely known BERT [3, 1] model which, also previously shown to achieve state-of-the-art results in the intent detection task.

While intent classification is traditionally based only on the sentence level information, in other text classification tasks, such as Content Types [13], texts are often analyzed as a composition of units (clause level cue analysis). Token level intent detection was also previously explored by [7, 4, 12]. [12] used token level intent information for joint intent detection and slot filling. In their work, intent of the utterance is computed by voting from predictions at each token. In [4], token-based intent detection, used to deal with multi-intent utterances, is performed based on hidden states of BiLSTM cells, as well as a feed-forward network applied to the final state. In these papers, however, authors use either sentence level intents as labels for every token [12, 4] or a statistical method such as tf-idf to identify the keywords responsible for sentence intents [7]. Unlike these solutions, we demonstrate a different approach to token information, in which the tokens labeled with intentions are not identified as keywords [7], nor the sentence-level intent is assigned to every token. In our work we propose an annotation scheme where instead of labeling the whole sentence, human annotators identify the individual tokens responsible for the sentence sentiment.

3 Dataset

Traditional intent detection datasets such as ATIS [5] or Snips [2] contain only sentence level labels. Apart from that, they are not demanding using current state of the art methods due to large number of examples per intent and grammatically correct language in phrases, achieving results around of 99% accuracy. This is why for the purpose of evaluation of our method we created our own dataset of computer mediated customer-agent helpline conversations in the banking domain. This dataset contains real human-human conversations of customers with customer service agents on Facebook's Messenger in the Polish language. From the initial 28000 question-answer pairs, we selected 924 messages corresponding to frequently asked questions. Those questions were then paired with one of the 24 labels corresponding to user intentions. These intentions included e.g. asking about confirmation of money transfer, or the status of their application. The list of labels and their respected number of examples is

shown in the Table 1. On average 68% of the tokens in user utterance were labeled with one of the intentions. The detailed descriptions of intentions and their examples are shown in the appendix.

Table 1 No. of examples per intent in train and test split of the dataset.

Intent	train	test
300	26	7
unblocking access	97	24
deposit machine fee	29	7
double charge	34	9
payment confirmation	22	6
canceling an application	30	8
application malfunction	18	5
trusted profile	10	3
card malfunction	69	17
contact request	17	4
server malfunction	26	6
sessions	21	5
sms	31	8
application status	29	7
cdm funds posting	23	6
application processing time	32	8
cash withdrawal	12	3
IBAN/BIC/SWIFT	35	9
blocking card documents	21	5
helpline waiting time	27	7
change of personal data	42	10
card delivery time	25	6
change of phone number	49	12
thanks	14	3
Sum	**739**	**185**

All texts have been anonymized, that is parts of the statements have been concealed to avoid exposing the data of real-life customers. The anonymized information included phone numbers, names, web addresses, bank names, specific products and services, as well as mentions of other non-bank brands.

4 Our solution

In our solution, instead of labeling every message with a single label, we utilize token level labels for users intents. This approach is motivated by the fact that certain words contain stronger semantic properties with respect to users' intention. Traditionally, the user intention is predicted based on the final state of BiLSTM at the beginning of the network, the end of the sequence, or both. The relevant information can, however, be present in various parts of the sentence as shown in the Figure 1. While LSTM cells tend to model long-term dependencies well, they can still suffer from vanishing gradient [9]. With the loss function being calculated solely based on the last state of the network, while relevant information is present in the middle, a problem of training the network to recognize important features arises, which may results in lower accuracy. By contrast, using token level labeling loss function is calculated based on token level prediction, enabling training signals to be better propagated by the network. This approach can be seen as analogous to the auxiliary classifiers in convolutional neural networks [14].

Modern intent detection models also use transformer based architectures, the most popular one being BERT. In these systems, sentence level classification is based on the embedding of a special [cls] token. Token level embeddings are computed using a self-attention mechanism that models contextual relationships within the sentence. However, those architectures can also benefit form token level information in the intent detection task, as previously shown in [1].

4.1 Annotation scheme

Motivated by these facts, we introduce a token level annotation scheme for the intent detection task in which annotators were tasked to label a small part of each utterance in direct correspondence to user's intention, leaving out sentence parts irrelevant to the query:

- Each statement is assigned exactly one intention e.g. *how long do I have to wait for the application?*[application_processing_time]
- The chosen intention concerns the main topic of the conversation
- The scope of a tag covers the part of the statement that is specific to the intention. If the statement is complex and the client describes the reason for making contact in a few sentences then, unless otherwise impossible, the sentences were annotated in a way that helped to indicate the intentions in their context, e.g. *Hello, I would like to order an activation package. I created an account, I received an activation package via text, valid for 48 hours, but I was not able to activate it within 48 hours, hence the need to receive a new activation package. How can I order it?* [unlocking_access]

Examples of sentences and their annotations are shown in the Figure 1. Due to the fact that the corpus is a set of computer-mediated texts, it is characterized by an informal style and due to that several inconveniences were noticed during the tagging process. Full annotation scheme is shown in the appendix.

4.2 Limitations

In general, the tagged intention was a part of sentences or one sentence. However, there were texts in which the described problem could be noticed from the context of the statement, rather than its direct meaning, in which case a few sentences were marked. Some intentions were related to failures of various types. In this case, selecting the sentence: *are you having a malfunction* – did not indicate the type of malfunction and the next part of the statement should be marked: *I can't pay with card since a few hours ago, the ATM won't even read it.* It is also worth to mention that annotating the dataset with token-level labels requires additional work and therefore can reduce the ammout of labeled data in a given timeframe. The increased complexity in annotation scheme also lead to inconsistent annotations between users that can possibly result in reduced models performance (something we have yet to explore).

5 Experiments

In our experiments, we test various types of token-level annotations and their impact on intent detection accuracy. As a baseline solution we chose an annotation method where only the sentence is labeled with user intention and the are no additional labels for tokens. Next, we tested the annotation method used in [12, 4] where each token in the utterance is labeled with the utterance's intent. Finally, we tested human-made annotations where the annotator identified words responsible for the user intention. In those annotations, we tested three different label formats. The first one involved labeling each token as either relevant on irrelevant to the intention in a binary classification method. The second one included labeling relevant tokens with the sentence's intent class. The third method was based on BIO (beginning, inside, outside) labels. This format is visualized in Figure 2.

5.1 Models

Two different models were chosen for testing classifier accuracy, one based on the BERT [3] network and the second one based on the BiLSTM model. For BERT implementation, we chose the base multilingual model. In our experiments we fine-tuned the model for both sequence labeling and the classification task. During the training, each token was labeled in a corresponding format. We also used BERT's special [CLS] token for labeling the entire sentence. Token level embeddings were mapped to their labels using a fully connected layer with softmax activation function. A visualization of the BERT model with BOI annotation scheme is shown in the Figure 2. The second model we used for testing was based on the

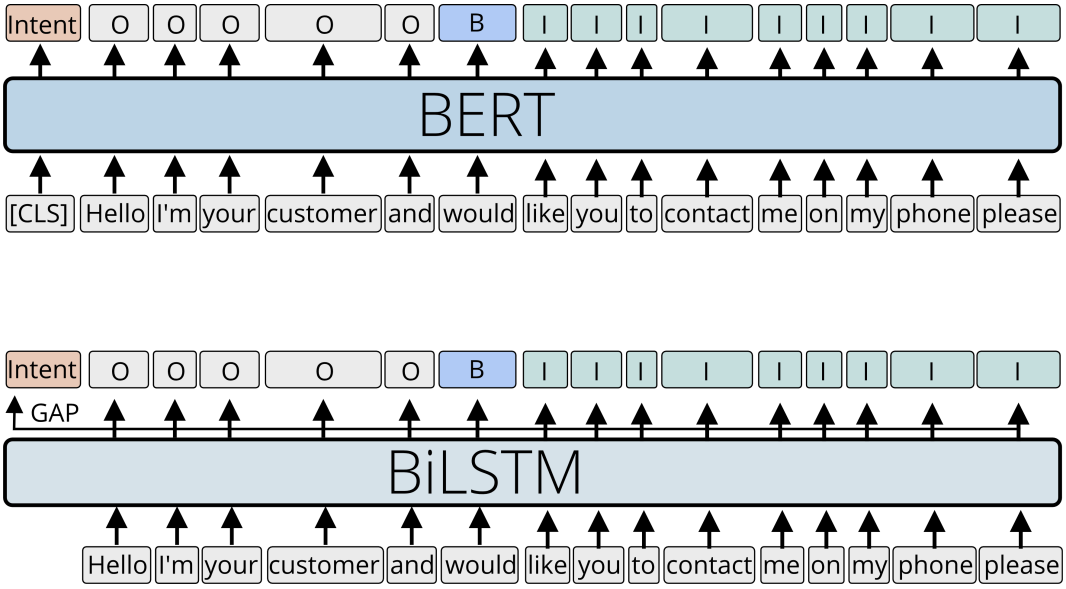


Figure 2 Examples of sentences and their annotation with token-level annotation method.

BiLSTM network. For the inputs we used Word2Vec embeddings pre-trained on the NKJP corpus [11]. These inputs were inputted into the bidirectional LSTM layer with a hidden state size of 300 neurons. Subsequently, for the token level classification we used a fully-connected layer with a softmax activation function. The sentence level labels were predicted based on LSTM cells output pooled with global average pooling, on top of which another fully connected layer with softmax activation function has been added.

5.2 Training

Both networks were trained using categorical cross entropy loss function. This loss was calculated between predicted token-level predictions and their true labels, as well as between sentence level intent prediction and its true intent. The loss function is shown in the Equation 1, where T is the number of tokens in the sentence, C_t is number of token classes dependant on the annotation style, C_s is the number of intents, t_i,p_i represent the correct token level class and prediction, and t_k and p_k represent sentence leave prediction and true class.

$$L = -\sum_{i}^{T}\sum_{j}^{C_t} t_i log(p_j) - \sum_{k}^{C_s} t_k log(p_k) \tag{1}$$

For network training we also used the Adam optimizer with a learning rate of 2e-5.

6 Results

Results of the models' accuracies using different token annotation methods are shown in the Table 2: not using token level annotations (no token labeled), using sentence intent as label for all the tokens (all tokens labeled), tokens labeled as either relevant or irrelevant to the sentence intention (binary labels), tokens labeled with the BIO scheme (BIO labels) and tokens relevant to the sentence intention labels with its intent (intent labels). We also compared our solution with baseline Support Vector Machines (SVM) model trained on the whole sentences without additional token labels.

Table 2 Comparison of the accuracies of BiLSTM and BERT models depending on different token level annotations.

Annotation scheme	BERT	BiLSTM	SVM
no tokens labeled	0.918	0.859	0.837
all tokens labeled	0.913	0.859	-
binary labels	0.918	0.864	-
BIO labels	0.929	0.864	-
intent labels	0.929	0.875	-

Results show that using token level annotations can boost the performance of a BERT based model by 1pp, while the BiLSTM model can raise it by 1.5pp. In both cases, the best results were achieved by labeling each relevant token with the utterance's intention. BERT model achieved an accuracy of 92.9 %, while BiLSTM achieved accuracy of 0.875 %. In contrast to the work presented by [12, 4], we have also determined that in our case labeling every token in the sentence does not improve the general accuracy of the system, and in the case of the BERT model worsens the outcome.

7 Conclusions and future work

In this paper we demonstrated that token level labeling can improve the accuracy of intent detection systems. In the future we are also planning on testing the influence of token level intent prediction on the accuracy of joint intent detection and slot filling models.

References

1 Qian Chen, Zhu Zhuo, and Wen Wang. Bert for joint intent classification and slot filling. *arXiv preprint*, 2019. `arXiv:1902.10909`.

2 Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint*, 2018. `arXiv:1805.10190`.

3 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint*, 2018. `arXiv:1810.04805`.

4 Rashmi Gangadharaiah and Balakrishnan Narayanaswamy. Joint multiple intent detection and slot labeling for goal-oriented dialog. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 564–569, 2019.

5 Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 753–757, 2018.

6 E Haihong, Peiqing Niu, Zhongfu Chen, and Meina Song. A novel bi-directional interrelated model for joint intent detection and slot filling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5467–5471, 2019.

7 Byeongchang Kim, Seonghan Ryu, and Gary Geunbae Lee. Two-stage multi-intent detection for spoken language understanding. *Multimedia Tools and Applications*, 76(9):11377–11390, 2017.

8 Bing Liu and Ian Lane. Attention-based recurrent neural network models for joint intent detection and slot filling. *arXiv preprint*, 2016. `arXiv:1609.01454`.

9 Christopher Manning and Richard Socher. Natural language processing with deep learning. *Lecture Notes Stanford University School of Engineering*, 2017.

10 Aleksander Obuchowski and Michal Lew. Transformer-capsule model for intent detection (student abstract). In *AAAI*, pages 13885–13886, 2020.

11 Narodowy Korpus Języka Polskiego. Nkjp. *bd). Pobrano*, 31, 2017.

12 Libo Qin, Wanxiang Che, Yangming Li, Haoyang Wen, and Ting Liu. A stack-propagation framework with token-level intent detection for spoken language understanding. *arXiv preprint*, 2019. `arXiv:1909.02188`.

13 Rachele Sprugnoli, Caselli Tommaso, Tonelli Sara, and Moretti Giovanni. The content types dataset: a new resource to explore semantic and functional characteristics of texts. In *15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017*, volume 2, pages 260–266. Association for Computational Linguistics, 2017.

14 Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

15 Congying Xia, Chenwei Zhang, Xiaohui Yan, Yi Chang, and S Yu Philip. Zero-shot user intent detection via capsule neural networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3090–3099, 2018.

16 Chenwei Zhang, Yaliang Li, Nan Du, Wei Fan, and Philip S Yu. Joint slot filling and intent detection via capsule neural networks. *arXiv preprint*, 2018. `arXiv:1812.09471`.

A Full Annotation Manual

A.1 Method of annotation

- Each statement is assigned exactly one intention e.g.
  how long do I have to wait for the application? - > [application_processing_time]
- The chosen intention concerned the main topic of the conversation.
- The scope of a tag covers the part of the statement that is specific to the intention
- If the statement is complex and the client describes the reason for his contact in a few sentences, then, if it was not possible otherwise, sentences were marked that helped to indicate the intentions in their context, e.g. *Hello, I would like to order an activation package.* ***I created an account, I received an activation package via text, valid for 48 hours, but I was not able to activate within 48 hours, hence the need to receive a new activation package. How can I order it?*** *-> [unlocking_access]*

A.2 Problematic cases in annotation

Due to the fact that the corpus is a set of computer-mediated text, it is characterized by an informal style and during the tagging process several inconveniences were noticed. Generally, the tagged intention was part of a sentence or one sentence. However, there were texts where the described problem could be noticed from the context of the statement, not from its directness, in which case a few sentences were marked. Some intentions were related to failures of various types. In this case, selecting the sentence: are you having a malfunction – it did not indicate the type of malfunction and the next part of the statement should be marked. e. g. *I can't pay with card since a few hours ago, the ATM won't even read it.*

A.3 Considered intentions

- Application_status – intention indicating the question about the status of the application
  - Hello, I would like to know **at what checking stage is currently my application**
  - I made a verification transfer yesterday, but I **still have no information about my application.**
- Payment_confirmation – intention indicating the question for confirmation of the transfer
  - I made a transfer and **cannot download the confirmation**
  - Good morning, **can I get a transfer confirmation** after the transfer has already been sent and the "send transfer confirmation to e-mail" box has not been checked?
- Blocking_card_documents – intention indicating the question about the possibility of blocking documents in the event of loss or theft
  - Hello, **please block my account urgently**. I am in Belarus at the moment and I do not use the card. This is some kind of theft. **Can I block the card somehow?**
  - Good evening. **I would like to block my account card. Can someone help me?**
- Trusted_profile – the intention indicating the question to create a profile trusted via the bank
  - Good morning. **I would like to set up a trusted profile** so that I can run errands in government offices. I would like to validate my profile through my bank account.
  - Good morning . I have a question: **is it possible to set up a TRUSTED PROFILE** in your bank, of which I am a customer?
- Sms – intention indicating the question about the problems associated with the coming message, codes, confirmations by text messages
  - Cool, you can't make transfers at this time, **the text hasn't arrived after over an hour**. It's not the first time either, ugh.
  - Hello, I have a problem with online transactions, **I am not getting any reply messages with the phone code**, what could be the reason?
- 300 – intention indicating the question about information on completing and submitting applications for the 300+ benefit
  - hello **how to apply for the "good start" benefit** through an account in your bank best regards
  - PLEASE TELL ME **HOW CAN I APPLY FOR 300 PLU** THRU THE BANK ??
- Canceling_an_application – intention indicating question about resign from the submitted application
  - **Can I cancel my application**? Unfortunately, the examination is taking too long and I cannot wait this long.
  - Hello, I would like to know **if I can cancel the loan application** to purchase goods from NonBankBrand

- Iban_bic_swift – intention indicating questions about IBAN and BIC/SWIFT numbers
  - Hello, **Where can I check the IBAN number and the BIC/SWIFT code** (?) ? Thank you for your answer in advance and have a nice day. FIRSTNAMETAG SURNAMETAG.
  - Hello, I have a foreign currency account at your bank and I would like to ask **what's the swift/bic number**?
- Card_malfunction – intention indicating questions related to card payment or cash withdrawal problems
  - Well, **my card has been rejected** 2 times when paying contactless and 2 times during a transaction with a card reader while using the correct pin
  - Good day. I can't get through to you ... **I have a problem with my card**. I do not know what's going on. **I cannot do payments or withdraw cash**. I can make transfers with no problems.
- Deposit_machine_fee – intention indicating question about the fees associated with use of machine deposit
  - Good morning, I have a personal account, **do you charge a fee for depositing money in a cash deposit machine**?
  - Hello. I have a CardBank debit card. **Is there any fee for using a cash deposit machine**?
- Thanks – intention indicating thanking
  - **thanks for help**
  - **thanks for the quick help**
- Sessions – intentions indicating the question related to sessions and transaction time
  - Hello, my friend made a transfer from NameBank at 12, I have an account at NameBank, **incoming sessions at NameBank are at 11:00, 15:00 and 17:00 the transfer should be here right**? and I did not receive the transfer, I contacted my bank but they said to contact you, I did not receive a transfer from you. it was made at 12. and no later
  - Hey, if I'm abroad, specifically in the Netherlands, and made a weekend transfer to another bank – NameBank from a PLN account to a PLN account, **is the posting time for such an operation extended? I've been waiting for the confirmation of the transfer since yesterday and I am starting to wonder if the funds will be delivered on time**. Today at the latest
- Helpline_waiting_time – intentions indicating the questions about hotline hours and connection waiting times
  - I have been blocked from accessing my account via the website. **I've been trying to call you, but for a long time no one has bothered to answer it... and you're supposed to be available 24/7...**
  - Hello, **I tried to connect with a consultant several times today and nobody's answering**... Please contact me
- Cash_withdrawal – intention indicating the question about withdrawing money at bank or atm
  - Hello. I have a question. **Will there be no problems if I go to your bank office tomorrow with the intention of withdrawing several thousand euros from my account**?
  - Hello, I have a small question **can I withdraw money from my account in any NameBank office in Szczecin**. I'm talking about a sum larger than what you can withdraw from an ATM

- Contact_request – intention indicating contact request
  - Hello again. Yes, **please have an expert contact me on my phone**. As soon as possible. I didn't manage to connect with an online expert and, to be honest, I am put off by this application. Please call me
  - Hello, I'm your customer and **would like you to contact me on my phone please**
- Card_delivery_time – intention indicating question about time of card delivery
  - Thank you for the information. I ordered the card through the application. **Please tell me how long is the waiting period for a new card**?
  - Hello. **How long does it take to get a multi-currency card**? And what are the account maintenance fees?
- Application_processing_time – intention indicating question about time of application processing
  - Hello, I would like to ask **how long will it take to process an application for a brokerage account**? Is it a matter of hours or days?
  - Hello, I would like to know **why it's taking so long to process a loan application**, it's been nearly 12 hours and I still haven't received a reply, while usually it would take a few to a dozen or so minutes. The application number is OTHERTAG
- Cdm_funds_posting – intention indicating question about posting funds via CDM
  - Hello. I **deposited money into the cash deposit machine** because I have to make an urgent transfer. The deposit was made at 20.03. **When will the money be on my account**?
- Server_malfunction – intention indicating questions regarding problems with the working of the website
  - Hello. **Why is it impossible to reach the WWWTAG website since Saturday's technical break**? the problem persists on many devices and with various internet providers I have already tried to reach your website on 4 devices and using 3 internet providers and nothing happened
  - Good evening. I have a question for you – **why isn't the NameBank website working and consequently it's impossible to log in to the account**
- Double_charge – intention indicating problems with double charge with paying by card
  - Good morning. I'm having an issue with a payment, so **my account has been double charged for the payment, how do I solve this problem**?
  - Good morning **Regarding yesterday's malfunction, will the payments that have been rejected be returned to my account**? They're still in the blocked and suspended tabs.
- Unlocking_access – intention indicating problems with logging
  - Hello. I have a question? **Is it possible to retrieve the password to the NameBank website**.
  - Hello. I would like to apply for a credit card, but **I don't remember my ID and password**. How do I solve this problem? Thank you in advance. Solution
- Change_of_personal_data – intention indicating questions about changing personal data in the system
  - good morning, my ID card has expired, I've gotten a new one. **Should I go to the bank office to update the data**?
  - Hi, **can I change my registered address I gave on the helpline**? While I've been creating an account?
- Application_malfunction – intention indicating question related to an application problem
  - Hello. T**he application won't work all day long**. Will it be up today?
  - Good morning. **I haven't been able to log into the NameBank's mobile application for several hours**. No connection message

- Change_of_phone_number – intention indicating question related to changing user's phone number.
  - Hi, **where can I change my phone number?**

A.4 Anonymization

All texts have been anonymised, which means that the names and parts of the statements have been hidden, on the basis of which the data and the person who concerns them can be recognized. The following were anonymised:

- address – cities, streets, addresses
- bank – bank names
- bankProduct – names of accounts, helpline, names of products related to the bank's brand
- cardBank – card names: visa, masterCard etc.
- FirstName – customer names
- other – application number, document number
- phone – phone numbers
- secondName – customer surnames
- user – nicknames, initializing clients
- www – pages, links
- nonBankBrand – ATMs, online stores and other stores, etc.

Towards Scope Detection in Textual Requirements

Ole Magnus Holter
Department of Informatics, University of Oslo, Norway

Basil Ell
Department of Informatics, University of Oslo, Norway

Abstract

Requirements are an integral part of industry operation and projects. Not only do requirements dictate industrial operations, but they are used in legally binding contracts between supplier and purchaser. Some companies even have requirements as their core business. Most requirements are found in textual documents, this brings a couple of challenges such as ambiguity, scalability, maintenance, and finding relevant and related requirements. Having the requirements in a machine-readable format would be a solution to these challenges, however, existing requirements need to be transformed into machine-readable requirements using NLP technology. Using state-of-the-art NLP methods based on end-to-end neural modelling on such documents is not trivial because the language is technical and domain-specific and training data is not available. In this paper, we focus on one step in that direction, namely scope detection of textual requirements using weak supervision and a simple classifier based on BERT general domain word embeddings and show that using openly available data, it is possible to get promising results on domain-specific requirements documents.

2012 ACM Subject Classification Computing methodologies → Natural language processing

Keywords and phrases Scope Detection, Textual requirements, NLP

Digital Object Identifier 10.4230/OASIcs.LDK.2021.31

Supplementary Material *Software (Source Code)*: `https://github.com/oholter/LDK2021_toward_scope_detection`; archived at `swh:1:dir:ea53d4ebc3fb613c158bcf53bc31a86e55f30e99`

Funding This research is funded by the SIRIUS centre: Norwegian Research Council project number 237898. It is co-funded by partner companies including DNV GL and Equinor.

1 Introduction

The number of standards, requirements, and recommended practices (RP) in industry is overwhelming and requirements administration is costly. A survey from 2016 reveals that costs related to requirements are particularly high for petroleum companies working on the Norwegian shelf [23]. There are international standards like ISO, IEC, national standards, industry standards and companies have their own set of standards and requirements and all of them are constantly evolving.

Most of the requirements that are used in industry today are available only in textual format embedded in documents (e.g., PDF, Word) and the documents are revised and treated as one entity. This way of dealing with requirements comes with numerous challenges. One challenge is the organization of the requirements (i.e., managing the documents). When a supplier agrees to build a specific artefact, he must know exactly what requirements are relevant for the task and to find them, he must look into many potentially relevant documents. The documents may contain duplicate, overlapping, and even conflicting requirements. The revision cycle time of requirements is also a major bottleneck for industry today. Before an international standard has been revised, the world has changed and they already seem outdated. This slow revision cycle is one of the main drivers of company-specific requirements which in turn creates a more complicated picture by multiplying the number of possible requirements, duplicates, and conflicts.

Some companies have migrated to requirement management systems such as SIEMENS' Polarion [30] and IBM's DOORS [21]. These tools allow for better organization of textual requirements. By adopting a proper numbering scheme where a requirement can be identified globally across documents and allowing metadata to be attached to a single requirement (e.g., author, comments), the requirements can be revised individually. This is a step in the right direction, but we are still dealing with requirements in natural language and not with machine-understandable requirements.

It would be of great help for industrial companies and requirements companies if the requirements were available in a structured, machine-understandable, format (e.g., RDF). That way, a machine will know what entity or process the requirement is about and also the relevant properties, conditions, and demands without having to figure it out by itself by interpreting the text. If we consider this requirement sentence: *Test pieces for transverse weld (cross weld) tensile shall be rectangular and in accordance with [B.2.3.3].* It can be broken down into its scope, and the demands:

- `Scope:` *Test pieces for transverse weld (cross weld)*
- `Demand 1:` *rectangular*
- `Demand 2:` *in accordance with [B.2.3.3].*

If every document is broken down into individual requirements and each requirement into its components, and if each of the components is linked with corresponding concepts in a knowledge base, requirements management can be done much more automatically because the requirements could also be used directly by computer applications. Revision cycles could be greatly reduced by not having to revise an entire document, it will be possible to detect duplicates and conflicts. We further imagine that relevant requirements for a particular project can be automatically identified, as well as which requirement belongs to which subset of the project. It can also be possible to do automatic compliance checking between a set of requirements and a project description. While new requirements can be made structural if industry decides on a standard model to represent requirements, we are still bound by over a century of textual requirements and standards that cannot be ignored.

Neural methods in NLP have improved performance on many NLP tasks. Such methods, however, require large amounts of training data. The pre-trained BERT-models, for instance, are trained using the English Wikipedia and the BookCorpus, in total about 3,300M words [16]. Working with neural NLP techniques on domain-specific documents is challenging. In the case when the available document base is small, it is not obvious how to obtain good-performing NLP-models. Different solutions have been proposed including distant supervision [25], domain specific word embeddings [26] and transfer learning approaches [29, 20]. Textual requirements documents are domain-specific, there is, to the best of our knowledge, no labelled data available, and not many publicly available knowledge bases.

Our key contribution is a method to label requirements sentences using a weakly supervised approach using openly available data and simple heuristics to train a classifier that classifies requirements sentences into **SCOPE** (contains a scope) and **NOT SCOPE** (the sentence does not contain the scope of the requirement). The source code is openly available.[1]

The remainder of this article is organized as follows. We define the problem in Section 2, before summarizing related work in Section 3. In Section sec:methodology we describe the creation of the corpus and the classifier; in Section 5 we evaluate the approach on real industry requirements before we discuss the findings in Section 6 and present our conclusions in Section 7.

[1] `https://github.com/oholter/LDK2021_toward_scope_detection`

2 The problem

Scope detection is the task of detecting a given requirement sentence's subject matter, i.e., the scope of the requirement. In this paper, we look at a binary classification task where a sentence is classified as containing a scope (**SCOPE**) or as not containing a scope (**NOT SCOPE**). According to the READI project [31, 32], a scope is typically a subclass of equipment. Thus, in this paper, we limit the task to the detection of a scope that is a piece of equipment, an assembly of pieces of equipment or refinement of equipment. A piece of equipment is understood as an artefact or a physical resource required for a purpose [5]. There can be requirement sentences with non-equipment scopes, such as about documentation, the outcome of events, quality of processes etc. However, we disregard such scopes.

We make two simplifying assumptions. First, that the scopes are in sentences with requirements that are strictly to be followed (mandatory requirements). According to [12], *shall* is a "verbal form used to indicate requirements strictly to be followed to conform to the document". For this reason, we disregard all sentences that do not contain the word *shall*. The second assumption is that the scope can be identified within a sentence, thus we disregard the surrounding context of the requirement sentence.

The requirement documents that we have been working with are structured documents using numbered sections, subsections and so on.[2] Each of the lowest level subsections can consist of either one or several sentences. Some examples of requirements sentences from the document DNVGL-ST-F101 (Submarine pipeline systems ed. October 2017) [12] are:[3]

- DNV-ST-F101 Sec.3 CONCEPT AND DESIGN PREMISE DEVELOPMENT
  3.3.4.5 *Marine growth on **pipeline systems** shall be considered, taking into account both biological and other environmental phenomena relevant for the location.*
- DNV-ST-F101 Sec.3 CONCEPT AND DESIGN PREMISE DEVELOPMENT
  3.3.6.1 *Surveys shall be carried out along the total length of the planned pipeline route to provide sufficient data for design and construction related activities.*
- DNV-ST-F101 Sec.3 CONCEPT AND DESIGN PREMISE DEVELOPMENT
  3.4.2.2 *The **submarine pipeline system** shall have a specified incidental pressure or be split into different sections with different specified incidental pressures.*
- DNV-ST-F101 Sec.4 DESIGN - LOADS
  4.2.1.7 *Fluctuations in temperature shall be taken into account when checking fatigue strength.*

The sentence in requirement **3.3.4.5** should be classified as **SCOPE** because it is about and contains a reference to *pipeline systems* which is an assembly of pieces of equipment. The sentence in **3.3.6.1** is a requirement for the planning of the pipeline route which is not a piece of equipment thus **NOT SCOPE**. **3.4.2.2** is **SCOPE** because it is about the submarine pipeline system and contains a reference to that. The sentence in **4.2.1.7** is **NOT SCOPE** since the scope is not explicit in the sentence (i.e., the fatigue strength of what?).

[2] The approach is not limited to documents with numbered sections and requirements.
[3] © DNV GL. DNV GL does not take responsibility for any consequences arising from the use of this content.

3 Related work

Scope detection of technical requirements is related to the classification of technical requirements in general. We consider four types of (sentence) classification on technical requirements: *i)* Demarcation of requirements from information, *ii)* detection of requirement defects (e.g., ambiguity, vagueness), *iii)* classifying requirements into a project's subsystem, and *iv)* classification of software requirements.

For the demarcation task, Winkler and Vogelsang propose to use a Convolutional Neural Network to classify sentences into either information or requirements using a dataset of automotive requirements that was manually labelled by industry partners [33]. Abualhaija et al. approach the same task by evaluating several statistical machine learning algorithms using a generic set of features and trained on labelled documents from different requirements domains [8]. For the detection of requirements defects, [28] uses rule-based NLP techniques (GATE). [19] tackles the project subsystem classification task by using a curated knowledge base extracted from technical specifications and a classification function, which takes into account the number of domain terms and context information. For the task of classifying software requirements using NLP techniques, a common task is the classification of software requirements into functional and non-functional requirements as in [15].

This work is also related to domain-adaption of NLP techniques such as transfer learning and weak supervision. One widely known form of weak supervision is distant supervision [24]. A typical application of distant supervision is relation extraction. To use distant supervision for relation extraction, we typically align a sentence with a database and whenever two entities are found in the sentence and there exists a relationship between the same entities in a dataset, the text is assumed to be about this relationship. Training data generated by distant supervision, however, can be noisy.

Snorkel [27] is a framework that aims to let the user programmatically build training data using a set of labelling functions. A labelling function is any code that aims to label a subset of input data. This can be simple heuristic rules or distant supervision using external knowledge bases. The user provides the framework with a set of labelling functions (LF). The framework is capable of learning a generative model to estimate the accuracy of the different labelling functions and combines the outputs into a set of probabilistic labels.

Another approach to domain adaption is transfer-learning [20]. Transfer learning means to use data or models from other tasks or domains to train a model for another task/domain [29]. An example of sequential transfer learning is to first pre-train word or sentence representations on a general corpus before fine-tuning them to the problem task/domain. Fine-tuning of pretrained BERT-embeddings for some downstream-tasks was evaluated in [16].

4 Methodology

We apply a pipeline of 5 major components as shown in Figure 1: *i)* Extraction of the textual content of the PDF *ii)* insert XML-tags to generate an XML-representation of the document, *iii)* extraction of the mandatory requirement sentences from relevant parts of the XML-document, *iv)* using data programming to create labelled training data, *v)* training of a classifier based on BERT pretrained contextual embeddings and a fully connected layer.

4.1 Notation

$\mathcal{R}$ is the set of all requirement sentences, r is a single requirement. $\mathcal{T}$ is a list of terms. **SCOPE** and **NOT SCOPE** are labels used to classify a requirement sentence as either containing the scope or not. **ABSTAIN** is used when a labelling function cannot decide.

Figure 1 Overview of the pipeline.

4.2 PDF to XML

Extracting text from a PDF document is not straightforward [14]. The requirements documents we are working with are structured using numbered headings for chapters, sections, subsection, tables, figures and so on. If we extract the text from the PDF using tools such as pdftotext and work with it directly, it is often difficult to identify sentence and paragraph boundaries. We find that by first converting the text to a document where the different elements of the document are surrounded by XML-tags, we can keep both the structure and the content of the document while having a document that is easier to parse for other applications.

We use Apache PDFbox [2] to extract the text from 52 requirements documents on rules for ship classification [10] and one on submarine pipeline systems [12]. Headers and footers were removed by restricting the area that was read by PDFBox.

First, we remove dots from common abbreviations (e.g., `Sec.` was changed to `Sec`, `Ch.` to `Ch`) to make it easier to identify sentence boundaries later. Then, symbols that are part of the XML-syntax (<,> and &) were changed (to <, >, and).

The start of sections and the various levels of subsections are identified using regular expressions and an XML-tag (e.g., <section>) is inserted in the text. The start of tables, figures and comments are found the same way (captions are above tables and figures in all the documents). Everything before the first section is removed. This is the introduction and the table of contents. Last, the XML preamble and the end document tag are inserted. End tags for each of the different tags (section, subsections, table, figure) are inserted when a new element of the same type starts, a new parent element starts, or where the document ends. Sometimes, tables in the document contain entries that are confused with section numbers or the document contained difficult structuring (e.g., first a table is within a note and then a note is within a table). In such cases, it is necessary to manually correct the XML file.

4.3 Extraction of requirements sentences

We parse the generated XML-file with a DOM parser and extract the textual content of each of the tags that contain requirement text. We do not use content from tables, figures and guidance notes. Each element is split, using the NLTK's recommended sentence tokenizer [6], into individual sentences. Finally, we collect all the sentences that contain the word shall (i.e., it is a mandatory requirement) and write them to a TSV file including the numbering of each of the sentences for reference.

4.4 Creation of labelled data

We use *snorkel* as a framework for creating a weak supervision-based training dataset. The framework is based on providing multiple labelling functions (LFs) that aim to classify a subset of the dataset. The framework unifies the output of the LFs and generates weak supervision labelled data. We have two labels **SCOPE**, **NOT SCOPE**. This is thus a binary document classification task. Each of our label functions works independently and, given a sentence, either classifies it into **SCOPE** or **NOT SCOPE** or abstains from classifying.

4.4.1 LF using dataset related to ISO 15926

Since we limit the approach to pieces of equipment, we use a gazetteer of artifacts. To compile this gazetteer, we used the SPARQL endpoint available at [1], which contains structured data according to the ISO 15926 standard. First, we queried for a list with all labels of concepts that are (recursively) subclasses of the ARTEFACT CLASS (`<http://data.15926.org/rdl/RDS201644>`). The result of this query was curated as follows: *i)* all strings were converted to lowercase, *ii)* the substring "class" was removed from the end of all class-names, *iii)* the substring "asme" was removed from the beginning of all class-names *iv)* if a string ends with "for asme ...", the substring starting with "for asme" to the end of the label was removed, *v)* duplicates were removed from the list.

Second, we queried for all concepts that are (recursively) subclass of ARTEFACT (`<http://data.15926.org/rdl/RDS422594>`). All strings were converted to lowercase. The two lists of terms were then concatenated into a final list of 10861 terms $\mathcal{T}_{15926}$. The labelling algorithm is described in Algorithm 1.

Algorithm 1 LF using dataset related to ISO 15926.

Input: $\mathcal{T}_{15926}, r$
Output: **SCOPE** or **ABSTAIN**

$\mathcal{C} \leftarrow$ get_noun_chunks(r)
for all $c \in \mathcal{C}$ **do**
    $s \leftarrow$ lemmatize_each_word(remove_stopwords(lowercase(word_tokenize(c))))
    while s not empty **do**
        if $s \in \mathcal{T}_{15926}$ **then return SCOPE**
        else $s \leftarrow$ remove_first_word(s)
return ABSTAIN

4.4.2 LF using list of words from TermoStat

We used the online TermoStat tool [17, 18] to extract domain vocabulary. We collected nouns both single and multi-word terms. We transformed each PDF-document into text using pdftotext and passed the text-documents one by one and collected all the terms in one list. The list was reduced by removing duplicates to 37,720 terms.

The list of domain vocabulary was filtered using WordNet. All terms that have hyponyms that are in the same synset as "artifact" were kept. This gave us a final list of 974 words, $\mathcal{T}_{term}$. The list includes words we probably wouldn't label as equipment, however, it is quite specific to the domain. Therefore it seemed best to create a function that, instead of label **SCOPE** all sentences that contain the word, it will label **NOT SCOPE** if after checking all chunks in the sentence, nothing was found. The labelling algorithm is described in Algorithm 2.

4.4.3 LF using list of words generated using word vectors

We used pre-trained word embeddings from gensim [3] (glove-wiki-gigaword-100). We manually compiled a small list of words (seed-terms) containing known pieces of equipment. The words were picked from ISO 14224 [22] and a few terms from [12] and [10] in total 30 words. The words are listed in Section A.2.

Algorithm 2 LF using list of words from TermoStat.

Input: $\mathcal{T}_{term}, r$
Output: **NOT SCOPE** or **ABSTAIN**
1: $\mathcal{C} \leftarrow$ get_noun_chunks(r)
2: **for all** $c \in \mathcal{C}$ **do**
3: $s \leftarrow$ lemmatize_each_word(remove_stopwords(lowercase(word_tokenize(c))))
4: **while** s not empty **do**
5: **if** $s \in \mathcal{T}_{term}$ **then**
6: $found \leftarrow True$
7: Break
8: **else** remove_first_word(s)
9: **if** $found == True$ **then return ABSTAIN**
10: **else return NOT SCOPE**

First, we calculated the average vector of these pieces of equipment. Then, we gathered the top 100 most similar terms (cosine similarity) to this vector and created a list $\mathcal{T}_{emb}$. For the labelling, we reused Algorithm 1, but with $\mathcal{T}_{emb}$ instead of $\mathcal{T}_{15926}$. We did not do any fine-tuning of the words used as seeds or of the vectors used, so it should be possible to improve upon this function.

4.4.4 LFs using simple regular expression patterns

We defined two label functions using simple regular expression patterns:

- If the sentence contains a colon: **NOT SCOPE** (often a definition)
  "Pressure, Maximum Allowable Incidental (MAIP): In relation to pipelines, this is . . . " is **NOT SCOPE**
- If it contains a capitalized "For": **SCOPE**
  "For full-lift safety valves, . . . " is **SCOPE**

If the sentence does not match the pattern, **ABSTAIN**.

4.4.5 LFs using simple terms

We also used two functions checking for terms in a sentence:

- Using a small manually created list of terms commonly seen in **SCOPE** sentences – see the Appendix.
- A manually created list of terms commonly seen in **NOT SCOPE** sentences – see the Appendix.

We did the most effort on compiling the list for the **NOT SCOPE** sentences because there is no dictionary that we can use to identify a sentence that does not contain any piece of equipment. The lists were created by manually looking at the requirements. Beginning with a few terms (e.g., report, carried out) we kept looking at the identified **NOT SCOPE**-sentences to identify terms that were common among them.

4.5 The classifier

To build the classifier, we use a pretrained BERT model [16, 4] (bert-base-cased) with a linear layer on top of the pooled output. For training, we use the values proposed in the original paper (and did no further tuning of hyper-parameters). The parameters are shown in Table 1. All parameters of the BERT model except bias, gamma and beta parameters are trained. The Linear scheduler with warmup is used.

Table 1 Hyper-parameters used to train the classifier.

Parameter	Value
epochs	4
learning rate	3e05
eps	1e-8
max seq len	32
optimizer	ADAMW
weight-decay-rate	0.01

5 Evaluation

5.1 Manual labelling

We created a labelling guideline for scope detection describing the task, the limiting conditions and example labelling of confusing cases. We used the guidelines and manually labelled 200 requirements sentences to evaluate performance when developing the dataset and the performance of the final model. When training the classifier, we also used 10% of the dataset for evaluation. Also, we divided the 300 sentences into three Excel sheets and asked ontology experts from DNV GL to annotate these sheets. Three annotators were selected for each sheet and they were given the same annotation guidelines that we had been using.

We further annotated 200 sentences from *Drilling facilities* (DNVGL-OS-E101) [9], all extracted sentences (180) from *Floating docks* (DNVGL-RU-FD) [11], and 200 sentences from Equinor's *Field instrumentation* (TR3032) [13].

5.2 Automatic labelling

The accuracy of the dataset with regard to the labelled dataset is 0.79 using *snorkel*'s majority vote model which was empirically found to give higher accuracy than *snorkel*'s label model (acc 0.76) for the task.

5.3 The model

The created training data was split into train and validation. The result of running the classifier on the validation dataset is found in Table 2. The accuracy of the classifier is 0.91. The result of running the classifier on the 200 manually labelled sentences is shown in Table 3 together with the results from the experiments with the three other documents, DNVGL-OS-E101, DNVGL-RU-FD, and TR3032.

Table 2 Result of the evaluation of the classifier on the validation set.

Class	Precision	Recall	F1	Support
NOT SCOPE	0.86	0.81	0.83	638
SCOPE	0.93	0.95	0.94	1801

[4] This is the document that the classifier was trained on.

Table 3 Evaluation of the classifier on manually labelled data from different documents.

Dataset	Class	Precision	Recall	F1	Support
RU-SHIP and ST-F101[4]	**NOT SCOPE**	0.87	0.64	0.74	92
Acc: 0.79	**SCOPE**	0.75	0.92	0.83	108
OS-E101	**NOT SCOPE**	0.80	0.47	0.60	76
Acc: 0.76	**SCOPE**	0.74	0.93	0.82	124
RU-FD	**NOT SCOPE**	0.60	0.46	0.52	68
Acc: 0.68	**SCOPE**	0.71	0.81	0.76	112
Equinor TR3032	**NOT SCOPE**	0.76	0.52	0.61	66
Acc: 0.79	**SCOPE**	0.79	0.92	0.85	134

5.4 Labelling by DNV GL

The manually labelled data from DNV GL were combined into one dataset using majority vote. Whenever an annotator had left a sentence unlabelled, we labelled it as scope if any of the two others had labelled it as scope. When there were missing values, however, it was not possible to calculate Kohen's kappa. For the second Excel sheet, we got only two of the three annotated sheets, so to combine the two sheets we labelled a sentence as scope if any of the two annotators had labelled it as scope. The accuracy of the classifier on the combined data was 0.67. Precision, recall and F1-score is shown in Table 4. We calculated inter-annotator agreement for each of the Excel sheets and the Kohen's kappa and Krippendorf's alpha scores are presented in Table 5 and Table 6 respectively.

Table 4 Evaluation of the classifier on the dataset labelled by ontology experts acc: 0.67.

Class	Precision	Recall	F1	Support
NOT SCOPE	0.84	0.48	0.61	162
SCOPE	0.59	0.89	0.71	138

Table 5 Kohen's kappa score between annotators.

Sheet no.	Annotators	Kohen's kappa
1	A B	-
	A C	0.335
	B C	-
2	C D	-
3	B E	0.582
	A E	0.368
	A B	0.408

Table 6 Krippendorff's alpha.

Sheet no.	Krippendorff's alpha
1	0.396
2	0.510
3	0.434

5.5 Examples of classification

The requirements sentences in this section are reproduced with permission from DNV GL.[5] Using the colour **olive**, we indicate the scope of the sentence. By the colour **red**, we indicate something that can be seen as a scope but is not considered in this paper.

5.5.1 Examples of correct classifications

- DNV-ST-F101 Appendix B MECHANICAL TESTING AND CORROSION TESTING
  B.2.3.9 [sentence 1] ***Test pieces for transverse weld (cross weld) tensile*** *shall be rectangular and in accordance with [B.2.3.3]* **SCOPE**
- DNV-RU-SHIP Pt.5 Ch.13 Sec.13 RADIATION HAZARDS
  4.1.2 [sentence 2]*Attention shall be paid to effects from re-radiated fields* **NOT SCOPE**
- DNV-RU-SHIP Pt.7 Ch.1 Sec.24 Winterized vessels
  24.2.1 [sentence 1] ***Anti-icing and de-icing switchboards*** *shall be surveyed.* **SCOPE**.
- DNV-RU-SHIP Pt.5 Ch.10 Sec.14 SLOP RECEPTION VESSEL **2.2.4** [sentence 2] *Coamings of suitable height shall be arranged below* ***manifolds*** *and* ***hose connections*** *in order to minimize spill.* **SCOPE**
- DNV-RU-SHIP Pt.5 Ch.10 Sec.6 DIVING SUPPORT VESSELS
  2.1.4 *If the diving support vessel does not carry the class notation COMF(V- crn),* ***the diver's accommodation area (inner area)*** *shall be subject to the relevant vibration and noise measurements applicable to the remaining accommodation.* **SCOPE**

5.5.2 Examples of incorrect classifications

In this section, we list some incorrect classifications and provide a short comment.

- DNV-RU-SHIP Pt.6 Ch.2 Sec.13 GAS FUELLED SHIP INSTALLATIONS - GAS FUELLED LPG
  9.3.3.2 [sentence 2] ***Detection of leakages*** *shall result in automatic closing of all valves required to isolate the leakage.* Falsely classified as **SCOPE**. Here, we don't know what we are detecting the leakage on. Fails possibly because a piece of equipment occurs in the sentence without being the scope.
- DNV-RU-SHIP Pt.4 Ch.3 Sec.2 GAS TURBINES
  2.2.5 [sentence 7] ***Burner*** *lifetime shall be specified together with the nominal/recommended exchange intervals.* Falsely classified as **NOT SCOPE**. It should be classified as **SCOPE** because lifetime as a direct property of burner.
- DNV-RU-SHIP Pt.7 Ch.1 Sec.2 ANNUAL SURVEYS EXTENT – MAIN CLASS
  3.3.1 [sentence 1] ***The survey on deck*** *shall include: – examination of the venting systems for the cargo tanks, interbarrier spaces and hold spaces.* Falsely classified as **SCOPE**. It is about the survey which is not a piece of equipment.

[5] © DNV GL. DNV GL does not take responsibility for any consequences arising from the use of this content.

- DNV-RU-SHIP Pt.5 Ch.10 Sec.14 SLOP RECEPTION VESSEL
  3.1.3 [sentence 2] – ***Two-way communication between the reception facility and the delivery vessel*** *shall be established before the transfer commences.* Falsely classified as **SCOPE**. It is about communication which is not a piece of equipment.
- DNV-RU-SHIP Pt.5 Ch.9 Sec.4 STANDBY VESSELS
  2.2.3 [sentence 1] *The net plate thickness, in mm, in* ***superstructures*** *and* ***deckhouses*** *shall not be less than: where: t0 = 4.5 for front bulkheads and weather deck forward of the lowest tier of the front bulkhead = 3.5 for sides and aft end bulkheads and weather decks elsewhere = 3.0 for superstructure and deckhouse decks (in way of accommodation) c = coefficient taken as:.* Falsely classified as **NOT SCOPE**.
- DNV-RU-SHIP Pt.6 Ch.1 Sec.4 STRENGTHENED FOR HEAVY CARGO IN BULK - HC
  10.1.2 [sentence 1] *Requirements specific to dry cargo ships The loading manual shall contain the loading conditions described in [4].* Falsely classified as **SCOPE**. It is about the loading manual, but the title seems to have been joined together with the requirement making it look like it is about dry cargo ships.
- DNV-ST-F101 Sec.7 CONSTRUCTION – LINEPIPE
  7.4.2.3 *In addition to the applicable information given in [7.1.7] and [7.1.8],* ***the MPS for lined linepipe*** *shall as a minimum contain the following information (as applicable): – details for fabrication of backing pipe and liner – quality control checks for the lining process – details of data to be recorded (e g expansion pressure/force, strain, deformation) – procedure for cut back prior to seal welding or cladding to attach liner to carrier pipe – seal welding procedures – details regarding any CRA clad welding to pipe ends.* Falsely classified as **SCOPE**. It is about MPS (manufacturing procedure specification) which is a document.
- DNV-ST-F101 Sec.6 DESIGN - MATERIALS ENGINEERING
  6.3.5.3 [sentence 2] *For* ***concrete coating*** *the minimum requirements in ISO 21809-5 shall apply with additional requirements given in [9.3] in this standard.* Classified as **SCOPE**, we regard concrete coating a material and classify the sentence as **NOT SCOPE**.

5.5.3 Sentence length and incorrect classification

Since our labelling functions identify terms in a sentence, it seems likely that we incorrectly classify longer sentences more often because they contain more terms. However, that this is not the case. The distribution of the incorrectly classified sentences is almost the same as the distribution of the entire dataset when plotted against sentence length as seen in Fig. 2.

6 Discussion

That majority vote gave better results than *snorkel*'s label function is in line with the results in [27] that when we have few labelling functions, many data points will have only one labelling function giving something different from **ABSTAIN**. From Table 7 in Appendix A, it is possible to see that the labelling functions that label **SCOPE** have better coverage than the ones that label **NOT SCOPE** and Table 8 show that the dataset performance is shifted toward higher recall on **SCOPE** and high precision and low recall on **NOT SCOPE**.

The detection of **NOT SCOPE** is in essence the detection of the lack of (technical) scope terms in a sentence. To detect the lack of terms we must have an exhaustive list of terms which we cannot assume to have. Having some terms that indicate **NOT SCOPE** (see the Appendix) is beneficial, but we suspect that many more patterns could be identified. It could be interesting to use clustering or frequent itemset mining to identify such terms.

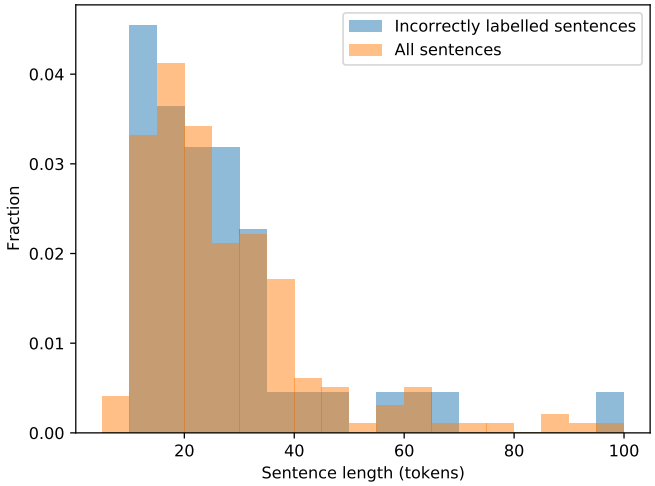


Figure 2 Distribution of the number of sentence per sentence length (number of tokens). The x-axis depicts the number of tokens in a sentence. The y-axis depicts the fraction of the total number of sentences.

For the word vector label function (Section 4.4.3) it would be interesting to try different sets of seed words and different word vectors. Using domain-specific word embeddings can also prove to be interesting.

The accuracy of the classifier is the same as using the dataset, 0.79. Taking a 95% confidence interval of the classifier's accuracy, we get 0.79 ± 0.0565 or $\approx [0.7335, 0.8465]$. The results of the classifier are encouraging. Having a classifier that can identify the relevant sentences for the scope detection task is a step toward detecting the scope of a requirement. As far as we know, the task of scope detection has not been tackled before in the literature.

We can see from the results in Table 3 that, as for the labelling functions, the model is also better at identifying **SCOPE** sentences than identifying **NOT SCOPE**. In general, recall is very high for **SCOPE** but rather low for **NOT SCOPE** sentences.

The evaluation of the approach on other documents shows promising results on both different topics and cross-company. The floating dock document shows the worst performance. This may be due to the vocabulary in this document that is not present in training data or the use of a more complex sentence structure. Somewhat surprisingly, the performance on Equinor's requirements document is comparable to the performance on the dataset from the same document the classifier was trained on.

We find, as expected that the model can falsely classify a sentence as **SCOPE** if the sentence contains equipment terms that are not scope. It does, however, also label many of them correctly. Sentences with quite complex scopes are also often classified correctly. We observe another challenge when the scope is a term that is followed by a non-scope-term as is the case for "burner lifetime". Several of the cases where the model disagreed with the manual labelling were also challenging to label manually and may be open for discussion.

We found no relation between the number of tokens in the sentence and misclassification (see Figure 2). It does, however, still seem that it has a higher probability of misclassification where the sentence contains equipment concepts that are not scopes. Some of the misclassifications are also very difficult for a human annotator. That may be because of the lack of context, ambiguous requirements or lack of domain knowledge.

For the manual labelling by the experts, we observe only moderate agreement among the annotators. We thought to clarify the annotation task by explicitly limiting the types of scopes to pieces of equipment. The boundaries of the equipment class can, however, be somewhat unclear. Thus, annotators did not agree on what is a piece of equipment in all cases. The lack of context information for each sentence also opens up for different interpretations by different annotators e.g., if a sentence says that an alarm shall sound if a certain condition is met, one annotator may consider that the sentence refers to the sound of the alarm and another annotator may consider it to be the physical entity that makes the sound. These results may also open up for discussions around some of the requirements sentences and how to create easier to understand requirements to better ensure a common understanding. To write clear and concise requirements is challenging and is the topic of [7].

Since a substantial amount of the sentences in the dataset does not contain a scope, training a classifier on the entire dataset to detect scopes would give a very imbalanced dataset and in a lot of sentences, it would not find anything. Manual labelling for scope detection would also be easier if most of the sentences actually contain a scope.

7 Conclusion

We have trained a model that classifies a requirement sentence into either **SCOPE** or **NOT SCOPE** depending on if the sentence contains the scope of the requirement or not. By using data programming we label a dataset with about 30.000 requirements sentences and train a simple BERT-based binary classifier on this dataset. The model shows good performance in separating sentences without a scope from sentences with scope with an accuracy of 0.79 on manually labelled sentences from the same document. The model also shows promising results on documents from other related domains and a document from another company. The performance of the model is, however, shifted toward high recall 0.92 for sentences with scope as opposed to a recall of 0.64 for sentences without scope.

By extracting the individual requirements from the document and splitting them into individual sentences, important contextual information is inevitably lost. Still, with this simplifying assumption, this seems like a promising first step toward scope detection of textual requirements. One major challenge for this task is that it is not trivial to ensure that all the involved parts have a common understanding of the problem, the sentences, and the terminology. The moderate inter-annotator agreements result even after having developed a guideline with many example sentences demonstrates this challenge.

References

1 15926browser. (visited on 2021-02-22). URL: `http://data.15926.org/rdl`.
2 Apache PDFBox | A Java PDF Library. `https://pdfbox.apache.org/` (visited on 2021-12-21).
3 Gensim. `https://radimrehurek.com/gensim/` (visited on 2021-02-17).
4 Google-Research/Bert. `https://github.com/google-research/bert` (visited on 2021-01-27).
5 Iso15926 equipment class. `http://data.15926.org/rdl/RDS8615020` (visited on 2021-02-22).
6 Natural Language Toolkit – NLTK. (visited on 2021-02-08). URL: `https://www.nltk.org/`.
7 INCOSE – guide for writing requirements, 2017.
8 S. Abualhaija, C Arora, et al. A Machine Learning-Based Approach for Demarcating Requirements in Textual Specifications. In *RE 2019*, pages 51–62, 2019.
9 Det Norske Veritas AS. Drilling facilities. Technical report, DNV-OS-E101, Ed. January 2018. © DNV GL.
10 Det Norske Veritas AS. Rules for classification: Ships. Technical report, DNV-RU-SHIP, Ed. July 2019. © DNV GL.

11 Det Norske Veritas AS. Floating docks. Technical report, DNVGL-RU-FD, Ed. October 2015. © DNV GL.

12 Det Norske Veritas AS. Submarine pipeline systems. Technical report, DNV-OS-F101, Ed. October 2017. © DNV GL.

13 Equinor ASA. Field instrumentation. Technical report, TR3032, Ver 3, August 2011. ©Equinor.

14 H. Bast and C. Korzen. A Benchmark and Evaluation for Text Extraction from PDF. In *2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, pages 1–10, 2017.

15 Agustin Casamayor, Daniela Godoy, and Marcelo Campo. Identification of non-functional requirements in textual specifications: A semi-supervised learning approach. *Information and Software Technology*, 52(4):436–445, 2010.

16 Jacob Devlin, Ming-Wei Chang, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv*, 2019. `arXiv:1810.04805`.

17 Patric Drouin. TermoStat Web. `http://termostat.ling.umontreal.ca/` (visited on 2021-02-17).

18 Patrick Drouin. Term extraction using non-technical corpora as a point of leverage. *Terminology*, 9:99–115, 2003.

19 G. Fantoni, E. Coli, et al. Text mining tool for translating terms of contract into technical specifications: Development and application in the railway sector. *Computers in Industry*, 124:103357, 2021.

20 Jeremy Howard and Sebastian Ruder. Universal Language Model Fine-tuning for Text Classification. *arXiv*, 2018. `arXiv:1801.06146`.

21 IBM. DOORS. `https://www.ibm.com/uk-en/products/requirements-management` (visited on 2021-03-08).

22 BS ISO. Iso 14224,"petroleum and natural gas industries: collection and exchange of reliability and maintenance data for equipment". *British Standards Institution, UK*, 1999.

23 Menon Economics. Requirements as cost drivers in the Norwegian petroleum industry. `https://www.menon.no/requirements-as-cost-drivers-in-the-norwegian-petroleum-industry` (visited on 2021-02-19).

24 Mike Mintz, Steven Bills, et al. Distant supervision for relation extraction without labeled data. In *ACL/AFNLP*, volume 2, pages 1003–1011, 2009.

25 Farhad Nooralahzadeh, Jan Tore Lønning, and Lilja Øvrelid. Reinforcement-based denoising of distantly supervised ner with partial annotation. In *DeepLo Workshop*, 2019.

26 Farhad Nooralahzadeh, Lilja Øvrelid, and Jan Tore Lønning. Evaluation of domain-specific word embeddings using knowledge resources. In *LREC 2018*, 2018.

27 Alexander Ratner, Stephen H Bach, et al. Snorkel: Rapid training data creation with weak supervision. *Proceedings of the VLDB Endowment*, 11(3), 2017.

28 Benedetta Rosadini, Alessio Ferrari, et al. Using NLP to detect requirements defects: An industrial experience in the railway domain. In *REFSQ*, pages 344–360, 2017.

29 Sebastian Ruder, Matthew E Peters, Swabha Swayamdipta, and Thomas Wolf. Transfer learning in natural language processing. In *NAACL Tutorials*, pages 15–18, 2019.

30 SIEMENS. Polarion REQUIREMENTS. `https://polarion.plm.automation.siemens.com/products/polarion-requirements` (visited on 2021-03-08).

31 SIRIUS. DREAM and READI: Cooperation to Manage Digital Requirements. `https://sirius-labs.no/dream-and-readi-cooperation-to-manage-digital-requirements/` (visited on 2021-03-08).

32 SIRIUS and DNV GL. On the READI method. Personal communication.

33 Jonas Winkler and Andreas Vogelsang. Automatic Classification of Requirements Based on Convolutional Neural Networks. In *RE Workshops*, pages 39–45, 2016.

A Additional details about the labelling functions

A.1 Coverage, overlap and performance

The coverage, overlap and conflicts among the labelling functions is presented in Table 7. We also include in Table 8 the performance of the dataset on the manually labelled sentences. It is important to notice, however, that these metrics are not directly comparable to the performance of the model as only considers sentences that are classified and ignores the **ABSTAIN** labels.

Table 7 Coverage, overlap and conflicts of the labelling functions.

Labelling function	Polarity	Coverage	Overlaps	Conflicts
has_equipment_termostat	**NOT SCOPE**	0.119	0.057	0.016
has_equipment_iso15926	**SCOPE**	0.523	0.355	0.138
has_equipment_wv	**SCOPE**	0.384	0.307	0.100
contains_colon	**NOT SCOPE**	0.105	0.094	0.082
contains_For	**SCOPE**	0.073	0.064	0.033
contains_scope_words	**SCOPE**	0.039	0.033	0.009
contains_non_scope_words	**NOT SCOPE**	0.215	0.170	0.124

Table 8 Precision, recall, f-score and support for the dataset on the manually labelled sentences disregarding all **ABSTAIN** labels (thus not comparable to the performance of the model).

Class	Precision	Recall	F1-score	Support
NOT SCOPE	0.87	0.65	0.74	71
SCOPE	0.79	0.93	0.85	100

A.2 List of seed words used for generating the word vector

The list of words used for generating the average word vector in 4.4.3: {“riser”, “accumulator”, “engine”, “compressor”, “blower”, “controller”, “generator”, “turbine”, “pipeline”, “sensor”, “pump”, “vessel”, “valve”, “bolt”, “cable”, “clamp”, “connector”, “cooler”, “fan”, “filter”, “fitting”, “flange”, “gearbox”, “joint”, “pipe”, “nut”, “pump”, “reflector”, “tube”, “ship”}

A.3 List of terms used for the LFs using simple terms

The words used for identifying **SCOPE** in Section 4.4.5: {“shall be capable of”, “shall be designed”, “shall be tested”}

The words used for identifying **NOT SCOPE** in Section 4.4.5: {“report”, “survey”, “shall include”, “describe”, “description”, “drawing”, “parameters”, “parameter”, “results”, “examination”, “include”, “include:”, “shall be taken as”, “shall be taken as:”, “carried out”, “shall be used to”, “shall cover”, “be based on”, “be performed”, “evaluate”, “calculation”, “calculations”, “analysis shall”, “criteria”, “based upon”, “determined”, “details’, “references”, “inspection”, “testing shall”, “hence”, “procedure”, “procedures”, “review”, “testing”}

Discrepancies Between Database- and Pragmatically Driven NLG: Insights from QUD-Based Annotations

Christoph Hesse
Leibniz-Zentrum Allgemeine Sprachwissenschaft, Berlin, Germany

Maurice Langner
Department of Linguistics, Ruhr-University Bochum, Germany

Anton Benz
Leibniz-Zentrum Allgemeine Sprachwissenschaft, Berlin, Germany

Ralf Klabunde
Department of Linguistics, Ruhr-University Bochum, Germany

Abstract

We present annotation findings when using an annotated corpus of driving reports as informational texts with an elaborated pragmatics for the automatic generation of corresponding texts. The generation process requires access to a database providing the technical details of the vehicles, as well as an annotated corpus for sophisticated, pragmatically motivated text planning. We focus on the annotation results since they are the basic framework for linking text planning with database queries and microplanning. We show that the annotations point to a variety of linguistic phenomena that have received little or no attention in the literature so far, and they raise corresponding questions regarding the access to information from databases for the generation process.

2012 ACM Subject Classification Applied computing → Arts and humanities

Keywords and phrases NLG, question-under-discussion analysis, information structure, database retrieval

Digital Object Identifier 10.4230/OASIcs.LDK.2021.32

Supplementary Material *Software (Corpus with annotated QUD-tree structures)*: `https://github.com/christoph-hesse/question-under-discussion`
Software (Annotation Tool): `https://github.com/MMLangner/QUDA`

Funding This work has been supported by the Deutsche Forschungsgemeinschaft, project "Propositional and Non-at-issue Content in Text Generation: Exploring the QUD–Perspective", grants AB 4348/5-1; KL 1109/7-1.

1 Generation (NLG) of pragmatically rich texts

Sufficiently precise annotations of linguistic phenomena in corpora are the backbone of almost every task in natural language processing (NLP) and generation (NLG). The interplay between theory-driven assumptions, the creation of annotation guidelines and the actual data analysis during the annotation process determine the quality of the annotation result.

In this paper, we are presenting our findings when annotating pragmatically motivated information structures in texts for the generation of driving reports. The generation process requires access to a database for retrieving information about the corresponding vehicles. For this, we may use the database of the ADAC, Germany's largest automobile club. However, driving reports do not only inform about technical details, but about subjective impressions and evaluations of the test driver as well, so that these texts are a mix of factual with subjective assertions. The gap between database retrieval and the presentation of subjective, evaluative information has – at least partially – to be closed by a learning approach that is not subject of this paper. Rather we show that even though the annotated information structures provide basic constraints for database retrieval, the annotation results point to a non-trivial match between annotation and retrieval.

The automatic generation of texts comprises several steps. It starts with providing the information to be verbalized from a data set and continues with the hierarchical ordering of the information (the text plan), the decision which information to realize in single sentences, ending with the language-specific grammatical tasks of determining the lexical items and grammatical encoding [5]. These individual tasks are not independent of each other, which holds even true for the first steps, the retrieval of information from a database and its arrangement in the text plan. If the task is not to generate a purely informational text but one which realizes pragmatically motivated information as, e.g., subjective estimations, attitudes and valuations, the text plan should include these kinds of information as well.

Typically, text plans are based on rhetorical relations for linking text spans in a coherent way [20]. However, rhetorical relations do not trigger constraints w.r.t the information structure in the respective text spans, and, from a theory-oriented point of view, due to their blurry definition, their explanatory power is rather limited.

We assume that question-under-discussion (QUD) approaches are more suitable for a theory-based construction of text plans. QUDs are the central concept in analyses that explain linguistic regularities as a consequence of the assumption that the sentences and text segments with which the regularities are associated are answers to an explicitly or implicitly asked question. QUDs figured prominently in theories explaining sequences of possible dialogue moves [3, 6], contextual relevance [16], information structural concepts (e.g. the *topic/focus* distinction, [16, 18, 19]), temporal progression in narration [9], and the analysis of coherence relations and subordination in text and dialogue [9, 18]. Although QUDs have been firmly established in theoretical linguistics, including the theory of discourse and dialogue, there has been, to our knowledge, only one attempt at developing guidelines and tools for text annotation [12, 15, 14], and no attempt at applying them in NLG systems. By generating driving reports, we aim at closing this gap.

In order to illustrate the annotation problem, let us have a look at a section on technical details in a German driving report about a motor bike. Driving reports are characterized by the fact that they combine factual information with subjective driving impressions and quality estimations. The constituents in that section that express information from a corresponding database are given in boldface.

1. **Eine voll ausgestattete Africa Twin mit elektronischem Fahrwerk kostet 18.665 Euro**. **Inklusive Liefer-Nebenkosten und Gepäcksystem** ist sie dann im Klub der **20.000-Euro-Reiseenduros** angekommen. Erste Probefahrten zeigen: Die Sache mit dem Fahrwerk ist kompliziert. Denn schon **das neue konventionelle Fahrwerk, in der Basis-Twin ausnahmslos verbaut**, **in der Sports Adventure regulär**, macht seine Sache ausgesprochen gut. Die für Letztere erhältliche **Option des elektronischen Fahrwerks (1600 Euro extra)** geht noch feinfühliger zur Sache, aber der Unterschied wird nicht für jedermann erfahrbar sein. Wir wetten: Die Topversion wird sich gut verkaufen.
   A fully equipped Africa Twin with electronic suspension costs 18,665 euros. **Including ancillary delivery costs and luggage system**, it then joins the club of **20,000-euro touring enduros**. First test rides show: The matter with the chassis is complicated. **The new conventional suspension, which is fitted without exception in the basic Twin, and as standard in the Sports Adventure**, does its job extremely well. **The electronic suspension option available for the latter (1600 euros extra)** is even more sensitive, but the difference will not be noticeable for everyone. We bet: The top version will sell well.

The constituents that are realizing these database-related information are unequally linked to different information structural levels. For example, *The new conventional suspension, which is fitted without exception in the basic Twin* expresses at-issue and non-at-issue information (the latter by the apposition), as a whole it could be a focus, and it contains a topic (*the suspension*). The annotations will shed light on the relation between facts and levels of information structure.

2 QUD-based annotation

In order to elaborate the relation between textual expansion and the realization of different kinds of information structures, we annotated 30 German driving reports based on the Question-under-discussion approach. Each QUD triggers a number of information structural decisions that together determine the realization of database-related information and subjective meaning components.

These QUD-dependent information structural distinctions are focus/background, topic/comment (with a further distinction between discourse topic and contrastive topic) and at-issue/non-at-issue information. The focus/background distinction as well as topic/comment received much attention in the literature, and their discourse-related function seems, at first, to be worked out in sufficient detail so that their annotation should be largely trouble-free. However, our annotation efforts point to a number of intricate problems in applying these categories to the respective clauses and text segments.

In what follows, we will explain the annotation of QUDs, focus/background, topic/-comment and at-issue/non-at-issue information according to the assumptions made in the literature and compare these guidelines with our insights from the annotation work. The consequences w.r.t. database retrieval for the generation process will be outlined.

2.1 The annotation process

Initially we were considering using the only existing annotation guidelines [15, 14] and tool [12] for this task, but we found two limitations that prompted us to develop our own: (1) The authors used QUD-tree-structures primarily for focus/background analysis, where focus constituents are taken to be answers to immediate super-QUDs, and background constituents are presupposed in the QUD. However, we were also interested in annotating topic/comment and evaluations through expressive, non-at-issue content. (2) The guidelines and the annotation tool allow for right-branching QUD-trees only.

We therefore developed our own XML-based annotation tool which allows for both right-branching and left-branching QUD structures.[1] In addition, our tool incorporates labeling of leaf node constituents for focus/background, topic/comment, at-issue/non-at-issue. It also incorporates indexing of constituents to capture phenomena such as split-focus/split-background where, for instance, a relative clause splits a focus constituent into a fragment before the relative clause and a fragment after it. An XML-based example is given below, together with its linguistic representation. Note that the QUD has been formulated by the annotator as a well justified question the sentence answers, and the information structures are derived from it.

```
<QUD String = "Was ist mit dem Antrieb?/What about the drive?">
    <F id="1"><SEGMENT> In der praktischen Außenhaut </SEGMENT></F>
    <CON>
        <SEGMENT> des 3,60 kurzen</SEGMENT>
        <SEGMENT>Fünftürers</SEGMENT>
    </CON>
    <F id="1"><SEGMENT> war der Antrieb </SEGMENT>
        <NAI><SEGMENT> erstmal </SEGMENT></NAI>
    <SEGMENT> kaum zu erkennen. </SEGMENT></F>
</QUD>
```

[1] The tool is available at `https://github.com/MMLangner/QUDA`.

2. [In der praktischen Außenhaut$_{F_1}$] des 3,60 m kurzen Fünftürers [war der Antrieb [erstmal$_{\mathrm{NAI}}$] kaum zu erkennen$_{\mathrm{F_1}}$]
   "In the practical outer skin of the 3.6 m short five-door car, the engine was hardly noticeable at first."

Our tool also allows the use of indexing for capturing focus/background and topic/comment distinctions where one constituent actually consists of multiple pieces of information (e.g., enumeration of facts), which have to be mapped to individual database entries.

Each driving report was independently annotated by two trained annotators (student assistants with a linguistic background). First, we split driving reports into sections (e.g., teaser, introduction, main sections which usually consists of technical specifications as specified by the manufacturer and test drive results either confirming the manufacturer's promise or not, available models of a vehicle and extras, and a summary/conclusion). Then, we proceed in a bottom-up fashion, annotating according to [15, 14] for focus/background, but also topic and non-at-issue. Above that we annotated common rhetorical structures such as *Contrast*, and above that structures which are more argumentative, for instance, why an author uses Contrast at this point (e.g., because of technical shortcomings of a vehicle, customer preferences/expectations or ongoing societal debates). Throughout this process, coreferential expressions (e.g., *der Fünftürer* [five-door car], *der kleinste Volkswagen* [the smallest VW], *der Stromer* [the electric], all referring to the same electric VW) are tracked, and indexing adjusted where needed. While QUDs on the bottom leaves are usually very concrete, the further up on the hierarchy QUDs are, the more abstract they become (e.g., QUDs for each section of a text, i.e., why certain technical details – e.g., opting for Diesel – are relevant given an issue raised earlier – e.g. Diesel Gate and the push for electric, thereby prompting *Contrast* comparisons with electric competitors).

2.2 QUDs

QUDs have to be formulated with database queries in mind. There are examples in our corpus of driving reports where this approach is easy enough to apply, and there are other examples where it is exceedingly more difficult. An easy example is an assertion about a car's acceleration, which should have a QUD such as *What's the car's acceleration?* with the focus constituent containing some measure of acceleration, e.g. *from 0 to 100 km/h in 5.7 seconds*, a measure phrase which should be generated from a database query about the car's acceleration. This approach is also easily applied to enumeration of facts, e.g., the different models available, which can be retrieved from the database through a QUD such as *What models are available?* This query would then return a list of size n, where each list element is a string. What makes QUD annotation more difficult is that authors of driving reports are often trying to maximise information density by speaking to multiple QUDs in the same proposition (e.g., what, what for, when, how, why QUDs), as in example 3:

3. Die Doppelstrategie ist aber gar nicht dumm, weil der Fahrer auf diese Weise sein spezielles Asphalt-Programm und ein ebenso spezielles Offroad-Programm hinterlegen kann.
   "However, this dual strategy is not dumb at all because it lets the driver choose their special/preferred (on-road) driving program and an equally special offroad program."

Annotators have to take extra care in deciding what the hierarchical relation between those QUDs is: Can they be put it in a clear hierarchy where authors are foregrounding some QUD and incrementally background a proposition's other QUDs, or are they equally important in light of a bigger argumentative point (e.g., that a release date may be unrealistic given engineering challenges or given societal pressures, or tastes of the target customer, or that expectations are contradicted as in example 3)?

In general, we want to be able to map the leaf nodes of our QUD trees to database entries, which puts constraints on the structure of those QUD trees. However, we find that QUD structures representing the more abstract levels of text interpretation show more variation across annotators given how they interpret the author's intent. A QUD tree structure which might do full justice to an author's nuanced argument might result in leaf nodes that are not easily mapped to the database, and vice versa, starting from leaf nodes that map easily to the database might necessitate a compromise higher up in the tree. We prioritize leaf nodes that easily translate to database queries. We made this choice not simply because a text generator would otherwise not be feasible, but because this also means that the resulting QUD structures towards the leaves of the tree can be evaluated in light of [12, 15, 14].

2.3 Focus/background

Focus and background are propositional attributes; the focus is that part of a proposition that is "new", i.e. put into the foreground. Its complementary part is the background. Foci as informational units correspond to specific syntactic constituents, the focus domains. Each focus domain contains a so-called focus exponent – the prosodically most salient element of this focus domain [7]. In this paper, we do not make reference to the phonological properties of foci, however. The relation between the semantic/pragmatic notion of focus and its linguistic counterpart, the focus domain, have been subject to a number of studies, as well as the rules and principles responsible for determining the focus exponent [13, 11, 10].

The default statement in linguistic semantics is to identify focus/background structures with two fundamental functions: They are marking information that is new for the listener, or they contrast information with already realized information (the so-called contrastive focus).

Foci are answers to a QUD; as such they provide new information. Equating focus with new and the background with given information, however, ignores the fact that often a given/new distinction can hardly be drawn, which also leads to corresponding uncertainty in the annotation process. As a consequence, one cannot directly link focus-annotated information with database queries for receiving the required new information in the generation process. Here are some examples for the non-trivial link between focus in our linguistic data and database access:

Split-focus: In our data, some sentences have two focus constituents that express one focus together. For example, one QUD in a driving report is *What about the power unit?* Example (2) provides the answer for this QUD. A plausible assignment of focus is to tag *In der praktischen Außenhaut* ("In the practical outer skin") and *war der Antrieb erstmal kaum zu erkennen* ("the power unit was hardly noticeable at first") as being focused, but not "the 3.60 short five-door car" since this constituent doesn't provide a part for the answer to the QUD.

A further but related phenomenon concerns sentences consisting of two coordinated main clauses, each with its own focus, but answering one QUD:

4. QUD: *How is the Renault Captur?*
   Der Renault Captur [wächst]$_F$ und [verändert seinen Charakter]$_F$.
   "The Renault Captur grows and changes its character."

It is reasonable to assume two separate foci since this coordination refers to two new aspects of the tested car. Both foci are well motivated by the QUD; they demonstrate that one QUD does not necessarily set up one focus only. Ellipses also indicate that the "one QUD – one focus" default can be violated:

5. QUD: *How have the aesthetics changed, compared to the old Captur?*
   [das sieht scharf trainiert]$_F$ und [angriffslustig aus]$_F$.
   "that looks sharply trained and ready to attack."

The non-elliptic sentence in German would be *das sieht scharf trainiert aus und das sieht angriffslustig aus*, with the prefix *aus* separated from the prefix verb *aussehen* and remaining in the base position, and the subject plus verb stem inserted in the second clause. The ellipsis forces an index as well for expressing that both foci belong together; otherwise the ellipsis cannot be handled correctly.

Without doubt, focused/new information is the information that must be retrieved from the database in order to present it to the user. However, as we have shown, the blurry distinction between given and new as well as the various partial mappings in our data between focus domains and foci do not allow for a unique retrieval process.

2.4 Topic/comment

Topics are discourse referents a statement – the comment – is made about. Linguistically, topic candidates are typically introduced by indefinite means and later on, they will be picked up by anaphoric expressions, resulting in a tree-like topic structure that describes what a section is about [8]. Topic structures might involve contrastive topics – discourse referents to whom discourse topics are compared. We are annotating both types of topics.

Topics are addressed by QUDs as well since they are mentioned in them. In order to retrieve the right information in the database, the topic referent must be given in it. As long as topic referents represent vehicles or vehicle parts, their annotations turned out to be easy. However, indexing for stating coreference of different topic expressions is sometimes unclear due to metonymy (see the examples above with reference to an electric VW).

2.5 At-issue/non-at-issue

Non-at-issue content is the part of an assertion that is optional in regard to the question under discussion, whereas at-issue is simply all relevant information given in the context. The optionality criteria is defined as the validity of the assertion as answer to the present QUD when the non-at-issue content is omitted. The lack of relevance which is implied by not being "at-issue" is limited to the context of the given QUD and does not entail the irrelevance of the presented information. According to [14], non-at-issue content itself denotes a different assertion including an associated subordinate QUD with a focus-topic distinction of its own, which is irrelevant in the context of the super-QUD in whose scope the constituents are not at-issue. Therefore, the annotation of non-at-issue is made more complex by the fact that depth and detailedness of the annotation decides how well non-at-issue can be distinguished in the respective context, and the identification of what is at-issue greatly depends on the choice of the QUD.

Non-at-issue content ranges from evaluative adverbs on sentence level to less obvious elements like embedding matrix clauses that name the source of a tradicted information, e.g., "[they say that$_{NAI}$][the [car$_F$] is [overpriced$_C$]]".

Retrieving information contained in non-at-issue from databases is highly complex. Evaluative adverbs, for example, mirror inferences and subjective impressions the author made on the basis of the propositional content. In example (6) technological understanding of relations between gas consumption and weight triggers evaluating the propositional content and expressing it as non-at-issue content. Therefore, this sort of content cannot simply be queried from a database but must be inferred from domain knowledge.

6. Surprisingly, the new Kawasaki consumes 4 litres fewer gas despite its 5% higher weight in comparison to the previous generation of this model.

The database we are using contains marks for different criteria of the cars, e.g. economic factors or build quality. These marks range from 1.0 (best) to 6.0 (worst) and were either calculated on the basis of sensor inputs (e.g., real gas consumption) or a set of rules. The annotation of non-at-issue in the corpus allows for the association of the evaluative adverbs with the marks given in the database, which provides the opportunity of predicting and probabilistically determining the usage of evaluative adverbs given the dataset from the database. [17] suggests different criteria for identifying non-at-issue content, which base on the observation that their content "survives under negation and projection".

Annotators need to pay much attention to the applicability of these criteria when formulating QUDs and identifying non-at-issue.

3 Evaluation of the annotation results

As [1] point out, sources of representation problems in annotating corpora are ambiguity (several possible tags for one linguistic entity), variation (several variants for one variable exist), uncertainty (no sufficient knowledge for an unambiguous annotation available), error (annotating incorrectly) and bias (using an unbalanced corpus).

QUD-oriented annotating is inherently faced with uncertainty as annotation problem. Only hints for identifying QUDs can be given. Hence, there is no fixed set of theoretically justified QUDs for a certain text type, which results in a wide range of plausible QUDs. Since the information structures we are interested in can directly be derived from the formulated QUD (focus, topic, (non)-at-issue), uncertainty will be propagated to theses annotation levels as well. An additional problem that might arise is ambiguity which becomes especially relevant when annotating focus, as our data show.

Furthermore, it is also difficult to compare our results with previous work on annotating information structure due to the varying linguistic complexity of the data, the coefficient used, the segment sizes for the tags used, and the different annotation guidelines controlling the annotation. Some exemplary studies shall illustrate this.

We primarily compared our QUD annotations with the results presented in [4] on QUD-based annotation of information structure, the only comprehensive QUD-based annotations we are aware of. Their data consist of sections of an English and a German interview which have been annotated by two trained annotators. Since the authors used Cohen's κ as coefficient, based on a flattened representation of QUD trees in a matrix, we adopted this approach in order to achieve comparable results.

Measuring the agreement of the QUD annotations has been performed by the authors by first mapping the QUD trees to a matrix that represents the segments spanned by the QUDs and then calculating Cohen's κ based on this matrix. The coefficients range between 0.45 and 0.53. The κ values for the information structural categories are acceptable or even robust (with the non-at-issue annotation having the highest values) with a negative outlier for contrastive topics.

We adopted this approach and achieved values between 0.45 and 0.78 for QUD annotations with a mean of 0.63, but without taking into account pre-theoretic heuristics as [4] did. For calculating κ values for information structural levels, [4] defined heuristic rules to prevent disagreement due to theoretically unclear cases. For example, all pronouns shall be annotated as background, and discourse connectors at the beginning of discourse segments are not annotated at all. We did not specify in the annotation guidelines how to annotate certain lexical items w.r.t. information structures. Instead all information structural decisions have to be derived from the respective QUD.

Since κ is sensitive to the units for which the statistic will be computed [2], we also computed the γ coefficient [21] for QUD annotations, since it especially allows us to measure for long spans of texts the categorization and unitizing as a joint task. In principle, Krippendorff's α is also a suitable coefficient for categorizing and unitizing, but in some constellations it is sensitive to segment length, while γ treats short and long segments in the same way [21, p. 463]. This feature is especially relevant for our discourse-related annotations. The value $\gamma = 1$ expresses that all annotators perfectly agree while $\gamma < 0$ signals that the annotation result is worth less than annotating at random.

The overall γ is 0.13. Non-at-issue annotations result in $\gamma = -0.075$, focus results in $\gamma = 0.115$, background in $\gamma = 0.08$, and the auxiliary tag for every constituent that cannot be assigned to one of the information structural notions results in $\gamma = 0.32$. These values show a considerable gap between the frequently used κ statistic and the rarely used γ; these results require a deeper analysis of the meaningfulness of applying these statistics to discourse phenomena.

However, in our γ statistic, for the structurally less important tags, agreement is three times higher than for topic, focus and non-at-issue. Further insights must be gained whether the combined text spans of the three main tags correspond, which would mean that the low γ agreement is due to confusion in the classification task. In general, QUD annotations are trees, which means that apart from the correct classification of the terminal nodes, the complexity of the tree structure needs to be compared as well. If the tree structures prove to be closely equivalent, this indicates that the low agreement is rooted only in the unique identification of topic, background and non-at-issue.

However, divergent discourse tree structures do not mean that the texts will be understood differently. Rather, what we observe is that the annotated structures express subjective opinions on the levels of information structure within texts, and structural decisions only weakly reflect subjective views.

4 Conclusion

The annotations show that retrieving information from a database for generating the text plan for driving reports involves an accurate annotation of focus, topic and non-at-issue information, since focused information should be retrievable from the database, and topic referents must be given as entities in the database as well. Non-at-issue information, often expressed by subjective estimations, require NAI-specific analyses of database entries in order to justify the use of corresponding linguistic means. For example, *überraschenderweise* "surprisingly" requires a comparison of the actual non-at-issue content with defaults in the database.

The annotation results contrasts with the high agreement concerning which facts should be realized. What the facts in the texts are is undisputed and what important facts are as well. We thus can state that the challenge is not to determine which facts to retrieve for their linguistic realisation, but how to do that on discourse level.

References

1 Christin Beck, Hannah Booth, Mennatallah El-Assady, and Miriam Butt. Representation problems in linguistic annotations: Ambiguity, variation, uncertainty, error and bias. In *Proceedings of the 14th Linguistic Annotation Workshop*, pages 60–73, Barcelona, Spain, 2020. Association for Computational Linguistics. URL: `https://www.aclweb.org/anthology/2020.law-1.6`.

2 Jean Carletta. Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics*, 22 (2):249–254, 1996.

3 Lauri Carlson. *Dialogue Games: An Approach to Discourse Analysis*. Reidel, Dordrecht, 1983.
4 Kordula De Kuthy, Nils Reiter, and Arndt Riester. QUD-based annotation of discourse structure and information structure: Tool and evaluation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, 2018. European Language Resources Association (ELRA). URL: `https://www.aclweb.org/anthology/L18-1304`.
5 Albert Gatt and Emiel Krahmer. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, 61:65–170, 2018.
6 Jonathan Ginzburg. Interrogatives: Questions, facts, and dialogue. In Shalom Lappin, editor, *Handbook of Contemporary Semantic Theory*, pages 359–423. Blackwell, Oxford, 1996.
7 Daniel Glatz and Ralf Klabunde. Focus as perspectivation. *Linguistics*, 5(41):947–977, 2003.
8 Zachary Kimo Stine and Nitin Agarwal. Comparative discourse analysis using topic models: Contrasting perspectives on china from reddit. In *International Conference on Social Media and Society*, SMSociety'20, page 73–84, New York, NY, USA, 2020. Association for Computing Machinery. `doi:10.1145/3400806.3400816`.
9 Wolfgang Klein and Christiane von Stutterheim. Quaestio und referentielle Bewegung in Erzählungen. *Linguistische Berichte*, 109:163–183, 1987.
10 Angelika Kratzer and Elisabeth Selkirk. New vs. given. In Daniel Altshuler and Jessica Rett, editors, *The Semantics of Plurals, Focus, Degrees and Times*, pages 157–162. Springer, 2019.
11 Manfred Krifka. Association with focus phrases. In Valerie Molnár and Susanne Winkler, editors, *The architecture of focus*, pages 105–136. Mouton de Gruyter, 2006.
12 Kordula De Kuthy, Nils Reiter, and Arndt Riester. QUD-Based Annotation of Discourse Structure and Information Structure: Tool and Evaluation. In Nicoletta Calzolari et al., editor, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France, May 2018. European Language Resources Association (ELRA).
13 Knud Lambrecht. *Information Structure and Sentence Form*. Cambridge University Press, 1994.
14 Arndt Riester. Constructing QUD trees. In Malte Zimmermann, Klaus von Heusinger, and Edgar Onea, editors, *Questions in Discourse: Pragmatics*, volume 2, pages 403–443. Brill, Leiden, 2019.
15 Arndt Riester, Lisa Brunetti, and Kordula De Kuthy. Annotation guidelines for questions under discussion and information structure. In Katharina Haude Evangelia Adamou and Martine Vanhove, editors, *Information Structure in Lesser-described Languages. Studies in prosody and syntax*, pages 403–443. John Benjamins, Amsterdam, 2018.
16 Craige Roberts. Information structure in discourse: Toward an integrated formal theory of pragmatics. In Jar Hak Yoon and Andreas Kathol, editors, *OSU Working Papers in Linguistics*, volume 49, pages 91–136. The Ohio State University, Department of Linguistics, Ohio, 1996.
17 Judith Tonhauser. Diagnosing (not-)at-issue content. In E. Bogal-Allbritten, editor, *Proceedings of the Sixth Conference on the Semantics of Under-represented Languages in the Americas and SULA-Bar*, Anherst, 2012. GLSA Publications.
18 Jan van Kuppevelt. Discourse structure, topicality, and questioning. *Journal of Linguistics*, 31:109–147, 1995.
19 Christiane von Stutterheim. *Einige Prinzipien des Textaufbaus: Empirische Untersuchungen zur Produktion mündlicher Texte*, volume 184 of *Reihe Germanistische Linguistik*. Niemeyer Verlag, Tübingen, 1997.
20 S. Williams and Ehud Reiter. Generating basic skills reports for low-skilled readers. *Natural Language Engineering*, 14 (4):495–525, 2008.
21 Antoine Widlöcher Yann Mathet and Jean-Philippe Métivier. The unified and holistic method gamma (γ) for inter-annotator agreement measure and alignment. *Computational Linguistics*, 3(41):437–479, 2015.

Bridging the Gap Between Ontology and Lexicon via Class-Specific Association Rules Mined from a Loosely-Parallel Text-Data Corpus

Basil Ell
CIT-EC, University of Bielefeld, Germany
Department of Informatics, University of Oslo, Norway

Mohammad Fazleh Elahi
CIT-EC, University of Bielefeld, Germany

Philipp Cimiano
CIT-EC, University of Bielefeld, Germany

Abstract

There is a well-known lexical gap between content expressed in the form of natural language (NL) texts and content stored in an RDF knowledge base (KB). For tasks such as Information Extraction (IE), this gap needs to be bridged from NL to KB, so that facts extracted from text can be represented in RDF and can then be added to an RDF KB. For tasks such as Natural Language Generation, this gap needs to be bridged from KB to NL, so that facts stored in an RDF KB can be verbalized and read by humans. In this paper we propose *LexExMachina*, a new methodology that induces correspondences between lexical elements and KB elements by mining class-specific association rules. As an example of such an association rule, consider the rule that predicts that if the text about a person contains the token *"Greek"*, then this person has the relation `nationality` to the entity `Greece`. Another rule predicts that if the text about a `settlement` contains the token *"Greek"*, then this settlement has the relation `country` to the entity `Greece`. Such a rule can help in question answering, as it maps an adjective to the relevant KB terms, and it can help in information extraction from text. We propose and empirically investigate a set of 20 types of class-specific association rules together with different interestingness measures to rank them. We apply our method on a loosely-parallel text-data corpus that consists of data from DBpedia and texts from Wikipedia, and evaluate and provide empirical evidence for the utility of the rules for Question Answering.

2012 ACM Subject Classification Computing methodologies → Information extraction; Computing methodologies → Natural language generation

Keywords and phrases Ontology, Lexicon, Association Rules, Pattern Mining

Digital Object Identifier 10.4230/OASIcs.LDK.2021.33

Supplementary Material *Collection (Dataset and Source Code)*: `http://www.LexExMachina.xyz`

Funding This work has been supported by the EU's Horizon 2020 project Prêt-à-LLOD (grant agreement No 825182) and by the SIRIUS centre: Norwegian Research Council project No 237898.

1 Introduction

There is a fundamental lexical gap between the "names", that is URIs, that are given to data elements in knowledge bases or knowledge graphs on the one hand, and how they are referred to in natural language. Bridging between these two symbol levels is crucial. There are many scenarios in which we need to map from natural language to KB, that is the case for text understanding, information extraction and question answering. There are also scenarios in which we need to map from KB to language, e.g. when verbalizing triples of a knowledge base in natural language [12].

In this paper, we present an approach to inducing correspondences between the lexical and knowledge base level that relies on mining association rules. The association rules that we mine have a lexical or linguistic symbol on the one side, and a KB symbol or structure on the other, thus allowing to bridge between the two levels.

The association rules that we mine are *class-specific* in the sense that at least one of the sides of an association rule expresses a condition that the entities that the association rule talks about belong to a specific class. The motivation for this is that the way a certain property is verbalized depends on the class in question. Similarly, the interpretation of a certain lexical element depends on the context of the class in question. Take the example of the adjective *Greek* that according to classical formal semantics represents a unary predicate, that is a class. When `Greek` modifies a person as in *"Greek politician"*, the correct interpretation with respect to the schema of a knowledge base might be the one that the `nationality` is `Greek`. In case of a city, e.g. *"Greek city"*, the correct interpretation might be that the `country` in which the city is located in is `Greece`. So the interpretation is class-specific. Conversely, take a property such as `author`. In the context of books, the property would be verbalized as *X wrote Y*, while in the context of a music piece the appropriate verbalization would be *X composed Y*.

In this paper we present our approach to mining class-specific association rules from a loosely-parallel dataset consisting of a corpus and corresponding knowledge base. The corpus and KB are loosely parallel in the sense that the text describes the entities in the KB but there is no explicit relation between the two. Further, the relation is not 1:1 in the sense that there are some triples that are not expressed in the text and there are many aspects in the text that are not represented by triples. We describe 20 different types of such class-specific association rules that we mine. We apply our approach to a parallel dataset consisting of the Wikipedia abstracts for 1,297,623 entities from 354 classes, together with the RDF descriptions of these entities. We derive 447,888,109 association rules from this dataset in total. We evaluate our approach on the basis of the well-known QALD (Question Answering over Linked Data) dataset, evaluating in how far our approach can retrieve valid correspondences between lexical and KB elements.

The remainder of this paper is structured as follows: we present our method for mining class-specific association rules in Section 2. We describe the application of our method on a loosely-parallel text-data corpus consisting of texts from Wikipedia and data from DBpedia in Section 3. We present the results of our evaluation on a question answering task in Section 4. Before concluding we discuss related work.

All code and data is available at our website `http://www.LexExMachina.xyz`.

2 Approach

In this section, we describe our approach *LexExMachina*. We introduce relevant preliminaries and notation needed to express the class-specific association rules in Section 2.1. We introduce our approach by an example describing a particular association rule for our motivating example in the introduction in Section 2.2. We describe our general approach in Section 2.3.

2.1 Preliminaries

Let P be a set of (URIs of) properties, let D be a set of documents, let C be a set of classes, let E be a set of entities, let G be an RDF graph, and let L be a set of linguistic patterns (for example, n-grams). Furthermore, let $c_e \subseteq C$ denote classes that entity $e \in E$ belongs to, let $d_e \in D$ denote the document that describes the entity $e \in E$ (e.g., the Wikipedia

article about the entity), and let $l_e \subseteq L$ denote the set of linguistic patterns that occur in the document d_e describing e. An RDF graph is a set of triples of the form (s,p,o) where $s \in \mathcal{U} \cup \mathcal{B}$ is called the triple's subject, $p \in \mathcal{U}$ is called the triple's predicate, and $o \in \mathcal{U} \cup \mathcal{B} \cup \mathcal{L}$ is called the triple's object. $\mathcal{U}$, $\mathcal{B}$, and $\mathcal{L}$ are the sets of URIs, blank nodes, and literals, respectively, and are pairwise disjoint. The set $\mathcal{T}$ of terms is the union of the sets $\mathcal{U}$, $\mathcal{B}$, and $\mathcal{L}$. The sets P, C, and E are true subsets of $\mathcal{U}$.

An association rule has the form $A \Rightarrow B$ where A and B are called events. For example, *Greece occurs in the text* is an event. The support of an event A, denoted by $sup(A)$, is the number of times that this event is true in a given set. For example, given a set of texts, the support of the event *Greece occurs in the text* is the number of documents for which it holds that *Greece occurs in the text*. The confidence of an association rule $A \Rightarrow B$, denoted by $conf(A \Rightarrow B)$, is defined as $conf(A \Rightarrow B) = sup(A \wedge B)/sup(A)$.[1] For example, let B be the event *born occurs in the text*. Thus, the confidence of the rule $A \Rightarrow B$ is the support of the event *Greece and born occur in the text* divided by the support of the event *Greece occurs in the text*. The higher the confidence, the more likely it is that given that a text contains the word *Greece*, it also contains the word *born*. Thus, the confidence of an association rule $A \Rightarrow B$ is identical to the estimated conditional probability $P(B|A)$.

In practice, association rules with high confidence do not necessarily disclose truly interesting event relationships [2]. Therefore, an *interestingness measure* quantifies the interestingness of an association rule. We list the classical null-invariant measures of interestingness as reformulated in terms of estimated conditional probabilities by Wu et al. [16] as well as the null-invariant measure *imbalance ratio* (IR), also introduced by Wu et al. [16]:

$$AllConf(A,B) = min\{P(A|B), P(B|A)\} \tag{1}$$

$$Coherence(A,B) = (P(A|B)^{-1} + P(B|A)^{-1} - 1)^{-1} \tag{2}$$

$$Cosine(A,B) = \sqrt{P(A|B)P(B|A)} \tag{3}$$

$$Kulczynski(A,B) = (P(A|B) + P(B|A))/2 \tag{4}$$

$$MaxConf(A,B) = max\{P(A|B), P(B|A)\} \tag{5}$$

$$IR(A,B) = \frac{|P(A|B) - P(B|A)|}{P(A|B) + P(B|A) - P(A|B) \times P(B|A)} \tag{6}$$

Note that all of these 6 metrics are symmetric, i.e., the order of the events A and B does not matter (e.g., $AllConf(A,B) = AllConf(B,A)$ for any events A and B). The estimated conditional probabilities can be calculated via support counts given the equations $P(B|A) = sup(AB)/sup(A)$ and $P(A|B) = sup(AB)/sup(B)$.

2.2 A Close Look at one Rule Pattern

In this section we describe a rule pattern with the name $c_s, l_s \Rightarrow po$ in detail, before we present the list of all 20 rule patterns in Section 2.3.

Given are a class $c \in C$, a property $p \in P$, a term $o \in \mathcal{T}$, and a linguistic pattern l. Given that an entity e is an instance of the class c and given that the linguistic pattern l occurs in the document d_e that describes the entity e, we want to predict whether the triple (e,p,o) is true. We define two events A and B. AB denotes the conjunction of these two events.

[1] In the remainder of the paper we write AB to denote $A \wedge B$.

$$A = c \in c_e \wedge l \in l_e$$
$$B = c \in c_e \wedge (e, p, o) \in G$$
$$AB = c \in c_e \wedge l \in l_e \wedge (e, p, o) \in G$$

Given a class $c \in C$ and a linguistic pattern l, the support of the event A, denoted by $sup(A)$, can be calculated as $|\{e \in E \mid c \in c_e \wedge l \in l_e\}|$ – thus, the support of the event A is the number of entities where each entity is an instance of the class c and where the linguistic pattern l occurs in the document that describes the entity.

Given a class $c \in C$, a property $p \in P$, and a term $o \in \mathcal{T}$, the support of the event B, denoted by $sup(B)$, can be calculated as $|\{e \in E \mid c \in c_e \wedge (e, p, o) \in G\}|$ – thus, the support of the event B is the number of entities where each entity is an instance of the class c and where the triple (e, p, o) exists in the graph G.

Given a class $c \in C$, a property $p \in P$, a term $o \in \mathcal{T}$, and a linguistic pattern l, the support of the event AB, denoted by $sup(AB)$, can be calculated as $|\{c \in c_e \wedge l \in l_e \wedge (e, p, o) \in G\}|$ – thus, the support of the event AB is the number of entities where each entity is an instance of the class c and where the linguistic pattern l occurs in the document that describes the entity and where the triple (e, p, o) exists in the graph G.

From these events we can construct association rules of the form $A \Rightarrow B$ given a class $c \in C$, a property $p \in P$, a term $o \in \mathcal{T}$, and a linguistic pattern l:

$$c \in c_e \wedge l \in l_e \Rightarrow (e, p, o) \in G$$

For example, with the class $c = dbo{:}Politician$, the property $p = dbo{:}nationality$, the term $o = dbr{:}Greece$, and the linguistic pattern $l = \text{"}Greek\text{"}$, we can create the following association rule:

$$dbo{:}Politician \in c_e \wedge \text{"}Greek\text{"} \in l_e \Rightarrow (e, dbo{:}nationality, dbr{:}Greece) \in G$$

Due to the fact that the linguistic pattern is a 1-gram, matching the pattern against a text is simple enough so that we can calculate the support of the three events via SPARQL queries.[2] Thus, we obtain the values $sup(A) = 128$, $sup(B) = 19$, and $sup(AB) = 19$. The confidence of an association rule of the form $A \Rightarrow B$ can be calculated as $sup(AB)/sup(A) = P(B|A)$. For our example, the confidence of the association rule is $sup(AB)/sup(A) = 19/128 \approx 0.15$.

If the class membership constraints are removed from the event definitions, then we obtain the events $A' = l \in l_e$ and $B' = (e, p, o) \in G$. For the example above, this results in the support values $sup(A') = sup(\text{"}Greek\text{"} \in l_e) = 58,563$, $sup(B') = sup((e, dbo{:}nationality, dbr{:}Greece) \in G) = 464$, and $sup(A'B') = sup(\text{"}Greek\text{"} \in l_e \wedge (e, dbo{:}nationality, dbr{:}Greece) \in G) = 445$, which results in the confidence value of $sup(A'B')/sup(A') = 445/58,563 \approx$

[2] $sup(A)$: `SELECT COUNT(?e) WHERE { ?e rdf:type dbo:Politician . ?e dbo:abstract ?a . FILTER (LANG(?a)="en" && REGEX(?a, "(^|\\ W)Greek(\\ W|$)")) }` → 128; $sup(B)$: `SELECT COUNT(?e) WHERE { ?e rdf:type dbo:Politician . ?e dbo:nationality dbr:Greece }` → 19; $sup(AB)$: `SELECT COUNT(?e) WHERE { ?e rdf:type dbo:Politician . ?e dbo:nationality dbr:Greece . ?e dbo:abstract ?a FILTER(LANG(?a)="en" && REGEX(?a, "(^|\\W)Greek(\\W|$)"))}` → 19. The parts before and after the term `Greek` ensure that the term either occurs at the beginning of the text or after a non-word character and that the term occurs either at the end of the text or is followed by a non-word character. The queries were ran against the public endpoint of DBpedia (`http://dbpedia.org/sparql`) on January 5, 2021.

0.0076, which is significantly lower than the confidence of the association rule *with* class membership constraints (i.e., ≈ 0.15). For this reason, in this paper we only investigate association rules that are class-specific. Note that if the word *Greek* appears in a text about a person, this might indicate that the person is of Greek nationality, whereas if the word *Greek* occurs in a text about a settlement, then this might indicate that the settlement is located in Greece – thus, which property is used depends on the class an entity belongs to.

If for an association rule $A \Rightarrow B$ we have calculated $sup(A)$, $sup(B)$, and $sup(AB)$, then we can not only calculate $P(B|A)$, but also $P(A|B)$, which means that we can calculate the confidence of the "reversed" association rule $B \Rightarrow A$:

$$c \in c_e \wedge (e, p, o) \in G \Rightarrow l \in l_e$$

The name of the reversed rule pattern $c_s, l_s \Rightarrow po$ is $c_s, po \Rightarrow l_s$. For the example above, this is the reversed rule:

$$dbo{:}Politician \in c_e \wedge (e, dbo{:}nationality, dbr{:}Greece) \in G \Rightarrow \text{"}Greek\text{"} \in l_e$$

The confidence of this rule (i.e., P(A|B)) is $sup(AB)/sup(B) = 19/19 = 1$. Given that for an association rule $A \Rightarrow B$ we have computed $P(B|A)$ and $P(A|B)$, we can also compute values for the interestingness measures. Note that because the interestingness measures are symmetric, the interestingness of the rule is the same as the interestingness of the reversed rule for this interestingness measure.

For the example above, with $P(B|A) = 19/128$ and $P(A|B) = 19/19$, we obtain the interestingness measurements $AllConf(A, B) \approx 0.15$, $Coherence(A, B) \approx 0.15$, $Cosine(A, B) \approx 0.39$, $Kulczynski(A, B) \approx 0.57$, $MaxConf(A, B) = 1$, and $IR(A, B) \approx 0.85$.

2.3 Class-specific association rule patterns

The complete set of 20 class-specific association rule patterns is shown in Table 1.

In the rules we have shown above the linguistic pattern occurs anywhere in a text. For the task of deciding whether a text expresses the triple (e_1, r, e_2), one typically regards the string between the mentions of e_1 and e_2 in the text. According to the principle of distant supervision [10], one assumes that a text expresses (e_1, r, e_2) if both entities are mentioned in the text. For example, for the property *dbo:author* the linguistic pattern that appears between the mentions of the arguments could be *is the author of* or *is best known for her*. Thus, we present rule patterns where the linguistic patterns that are made use of do not occur anywhere in a text but instead need to occur between the arguments of a relation. We refer to these rule patterns as *localized rule patterns* and to the rules where linguistic patterns can occur anywhere in the text as *non-localized* rule patterns. Note that because localization is predicate-specific, rule patterns that do not specify a predicate cannot be localized.

Let $l_e^{c,p,d}$ denote the set of linguistic patterns that occur in the document d_e that describes the entity e where e is an instance of the class c and where the linguistic patterns occur between the arguments of the relation p. The arguments of the relation appear in the order d, which is either *so* (*subject then object*), or *os* (*object then subject*).

The following localized rule predicts a property-object pair for an entity where in the text about the entity a linguistic pattern occurs that has been found between arguments of this relation in other text about entities of the same class:

$$dbo{:}Settlement \in c_e \wedge \text{"the Metropolitan City of Turin"} \in l_e^{dbo:Settlement,dbo:region,so}$$
$$\Rightarrow (e, dbo{:}region, dbr{:}Piedmont) \in G$$

Table 1 The list of 20 class-specific association rule patterns.

$c \in c_e \wedge l \in l_e \Rightarrow (e,p,o) \in G$	$(c_s, l_s \Rightarrow po)$
$c \in c_e \wedge l \in l_e^{c,p,d} \Rightarrow (e,p,o) \in G$	$(c_s, ll_s \Rightarrow po)$
$c \in c_e \wedge l \in l_e \Rightarrow \exists o \in \mathcal{T} : (e,p,o) \in G$	$(c_s, l_s \Rightarrow p)$
$c \in c_e \wedge l \in l_e^{c,p,d} \Rightarrow \exists o \in \mathcal{T} : (e,p,o) \in G$	$(c_s, ll_s \Rightarrow p)$
$c \in c_e \wedge l \in l_e \Rightarrow \exists p \in \mathcal{U} : (e,p,o) \in G$	$(c_s, l_s \Rightarrow o)$
$c \in c_e \wedge l \in l_e \Rightarrow (s,p,e) \in G$	$(c_o, l_o \Rightarrow sp)$
$c \in c_e \wedge l \in l_e^{c,p,d} \Rightarrow (s,p,e) \in G$	$(c_o, ll_o \Rightarrow sp)$
$c \in c_e \wedge l \in l_e \Rightarrow \exists p \in \mathcal{U} : (s,p,e) \in G$	$(c_o, l_o \Rightarrow s)$
$c \in c_e \wedge l \in l_e \Rightarrow \exists s \in \mathcal{U} \cup \mathcal{B} : (s,p,e) \in G$	$(c_o, l_o \Rightarrow p)$
$c \in c_e \wedge l \in l_e^{c,p,d} \Rightarrow \exists s \in \mathcal{U} \cup \mathcal{B} : (s,p,e) \in G$	$(c_o, ll_o \Rightarrow p)$
$c \in c_e \wedge (e,p,o) \in G \Rightarrow l \in l_e$	$(c_s, po \Rightarrow l_s)$
$c \in c_e \wedge (e,p,o) \in G \Rightarrow l \in l_e^{c,p,d}$	$(c_s, po \Rightarrow ll_s)$
$c \in c_e \wedge \exists o \in \mathcal{T} : (e,p,o) \in G \Rightarrow l \in l_e$	$(c_s, p \Rightarrow l_s)$
$c \in c_e \wedge \exists o \in \mathcal{T} : (e,p,o) \in G \Rightarrow l \in l_e^{c,p,d}$	$(c_s, p \Rightarrow ll_s)$
$c \in c_e \wedge \exists p \in \mathcal{U} : (e,p,o) \in G \Rightarrow l \in l_e$	$(c_s, o \Rightarrow l_s)$
$c \in c_e \wedge (s,p,e) \in G \Rightarrow l \in l_e$	$(c_o, sp \Rightarrow l_o)$
$c \in c_e \wedge (s,p,e) \in G \Rightarrow l \in l_e^{c,p,d}$	$(c_o, sp \Rightarrow ll_o)$
$c \in c_e \wedge \exists p \in \mathcal{U} : (s,p,e) \in G \Rightarrow l \in l_e$	$(c_o, s \Rightarrow l_o)$
$c \in c_e \wedge \exists s \in \mathcal{U} \cup \mathcal{B} : (s,p,e) \in G \Rightarrow l \in l_e$	$(c_o, p \Rightarrow l_o)$
$c \in c_e \wedge \exists s \in \mathcal{U} \cup \mathcal{B} : (s,p,e) \in G \Rightarrow l \in l_e^{c,p,d}$	$(c_o, p \Rightarrow ll_o)$

In this example, $l_e^{dbo:Settlement,dbo:region,so}$ is the set of linguistic patterns that occur in d_e and that frequently occur in texts about instances of the class *dbo:Settlement* between the arguments of the relation *dbo:region* where these arguments appear in the order *subject then object*.

3 Mining class-specific association rules from Wikipedia and DBpedia as loosely-parallel corpus

The loosely-parallel text-data corpus we use consists of seven files[3] from the English DBpedia [1]. We refer to it as a loosely-coupled text-data corpus because this data contains the short abstracts of Wikipedia articles as well as structured data extracted from DBpedia. The information that is contained in the DBpedia files has not been extracted from the article's natural language text, which means that not every piece of information contained in

[3] From `https://wiki.dbpedia.org/develop/datasets` we retrieved the following files in the stated versions: infobox-properties_lang=en.ttl.bz2 (v2020.11.01), instance-types_lang=en_specific.ttl.bz2 (v2020.12.01), mappingbased-literals_lang=en.ttl.bz2 (v2020.12.01), mappingbased-objects_lang=en.ttl.bz2 (v2020.12.01), short-abstracts_lang=en.ttl.bz2 (v2020.07.01), labels_lang=en.ttl.bz2 (v2020.12.01), and anchor-text_lang=en.ttl.bz2 (v2020.12.01). Labels and anchors were only used to identify the arguments of a relation so that localized linguistic patterns can be collected. That means that *rdf:type* and *rdfs:label* never occur as predicate in any rule that we have mined.

an article is contained in a DBpedia file. Furthermore, not every piece of information that is contained in a DBpedia file is expressed in a Wikipedia article's short abstract (e.g., an athlete's height is usually only contained in a table and is not expressed in the text).

By restricting a class to have at least 100 instances and ignoring *owl:Thing*, we obtained a set of 354 classes ($min\_entities\_per\_class = 100$). For each class, we randomly selected at most 10,000 instances ($max\_entities\_per\_class = 10,000$). In total, we selected 1,297,623 entities, which amounts to approximately 22.63% of all entities for which an abstract exists.

We tokenized the abstract of each entity by splitting at whitespaces and then removed the characters dot ('.'), comma (','), round brackets ('(' and ')'), and colon (':'). From the obtained token sequences we extracted those n-grams ($n \in [1..5]$) that contain at least one non-stopword – we used the NLTK stopword list,[4] which contains 127 entries. We discarded those 1-grams that consist of less than four characters ($min\_onegram\_length = 4$).

For the localized property patterns, we carried out a simple form of coreference resolution, replacing the pronouns *he*, *she*, and *it* with the entity's *rdfs:label*.

For patterns to be localized, the arguments of a relation need to be detected. For this purpose we make use of an entity's *rdfs:label* as well as those anchor texts that refer at least 10 times to a given entity ($min\_anchor\_count = 10$). We also try to identify literal values. We convert literals of type *xsd:date* into a natural language representation such as *2021-03-21*$\hat{}\hat{}$*xsd:date* to *21 March 2021*, but leave literals with other datatypes unchanged. If both arguments of a relation were detected and the length of the string between the arguments is not higher than 100 characters ($max\_propertystring\_length = 100$) and consists of at least 5 characters ($min\_propertystring\_length = 5$), we tokenized the string and extract n-grams ($n \in [1, 5]$) as described above. For each pattern, we recorded in which order the arguments occurred in the text (i.e., $d \in \{so, os\}$).

The set of linguistic patterns for a class is the set of all n-grams that were found for at least 5 instances of the class ($min\_pattern\_count = 5$). For the localized property patterns, a pattern had to occur for at least 5 instances of the class ($min\_propertypattern\_count = 5$) for each combination of class and property and order of arguments. This means that the rules have, depending on which side the linguistic pattern occurs, a value for $sup(A)$ or $sup(B)$ of greater or equal to 5.

Given the parameter settings above, we obtained 447,888,109 rules – 427,541,617 non-localized rules and 20,346,492 localized rules. The number of rules found for each rule pattern is shown in Table 2. Note that we set rather low threshold values as this allows to extract data for higher threshold values by filtering, instead of mining, and to find appropriate threshold parameters (e.g., for $sup(A)$, $sup(B)$, $sup(AB)$, $P(B|A)$, $P(A|B)$). For a particular linguistic pattern, i.e., the token "Greek", Table 3 shows the 20 localized rules that are ranked highest according to the Cosine interestingness measure. These rules contain the linguistic pattern on any side of the association rule.

4 Evaluation

We evaluate the utility of the rules that we have mined in the context of the task of Question Answering over an RDF knowledge base. Given a natural language question and an RDF knowledge base, typically, the goal is to infer a SPARQL query that represents the meaning of the question using the KB's vocabulary, so that evaluating the query on the KB results in the KB's answer(s) to the question. We created a corpus of (question, query) pairs from the

[4] The list of stopwords is available at `https://gist.github.com/sebleier/554280` (Accessed 2021-02-20).

Table 2 The number of rules found for each rule pattern.

Group of rule patterns		Number of rules
$c_s, po \Rightarrow l_s$	$c_s, l_s \Rightarrow po$	75,127,937 each
$c_s, po \Rightarrow ll_s$	$c_s, ll_s \Rightarrow po$	4,500,459 each
$c_s, p \Rightarrow l_s$	$c_s, l_s \Rightarrow p$	98,317,655 each
$c_s, p \Rightarrow ll_s$	$c_s, ll_s \Rightarrow p$	5,293,226 each
$c_s, o \Rightarrow l_s$	$c_s, l_s \Rightarrow o$	67,147,957 each
$c_o, sp \Rightarrow l_o$	$c_o, l_o \Rightarrow sp$	3,812,313 each
$c_o, sp \Rightarrow ll_o$	$c_o, ll_o \Rightarrow sp$	157,519 each
$c_o, s \Rightarrow l_o$	$c_o, l_o \Rightarrow s$	429,627 each
$c_o, p \Rightarrow l_o$	$c_o, l_o \Rightarrow p$	6,499,288 each
$c_o, p \Rightarrow ll_o$	$c_o, ll_o \Rightarrow p$	222,042 each
		447,888,109 total

Table 3 The top-20 localized rules that contain the linguistic pattern *Greek*, ordered by the Cosine interestingness measure. We abbreviated *dbo*:$FormerMunicipality$ to *dbo*:FM.

Cos	Rule
0.9	$dbo{:}Model \in c_e \wedge (e, dbp{:}birthPlace, dbr{:}Greece) \in G \Rightarrow$ "Greek" $\in l_e^{dbo:Model,p,so}$
0.9	$dbo{:}Model \in c_e \wedge$ "Greek" $\in l_e^{dbo:Model,p,so} \Rightarrow (e, dbp{:}birthPlace, dbr{:}Greece) \in G$
0.88	$dbo{:}RugbyClub \in c_e \wedge (e, dbo{:}location, dbr{:}Greece) \in G \Rightarrow$ "Greek" $\in l_e^{dbo:RugbyClub,dbo:location,so}$
0.88	$dbo{:}RugbyClub \in c_e \wedge$ "Greek" $\in l_e^{dbo:RugbyClub,dbo:location,so} \Rightarrow (e, dbo{:}location, dbr{:}Greece) \in G$
0.88	$dbo{:}Model \in c_e \wedge (e, dbo{:}birthPlace, dbr{:}Greece) \in G \Rightarrow$ "Greek" $\in l_e^{dbo:Model,dbo:birthPlace,so}$
0.88	$dbo{:}Model \in c_e \wedge$ "Greek" $\in l_e^{dbo:Model,dbo:birthPlace,so} \Rightarrow (e, dbo{:}birthPlace, dbr{:}Greece) \in G$
0.87	$dbo{:}FormerMunicipality \in c_e \wedge (e, dbo{:}country, dbr{:}Greece) \in G \Rightarrow$ "Greek" $\in l_e^{dbo:FM,dbo:country,so}$
0.87	$dbo{:}FM \in c_e \wedge (e, dbo{:}type, dbr{:}Prefectures\_of\_Greece) \in G \Rightarrow$ "Greek" $\in l_e^{dbo:FM,dbo:country,so}$
0.87	$dbo{:}FM \in c_e \wedge (e, dbp{:}subdivisionName, dbr{:}Greece) \in G \Rightarrow$ "Greek" $\in l_e^{dbo:FM,dbp:subdivisionName,so}$
0.87	$dbo{:}FM \in c_e \wedge$ "Greek" $\in l_e^{dbo:FM,dbo:country,so} \Rightarrow (e, dbo{:}country, dbr{:}Greece) \in G$
0.87	$dbo{:}FM \in c_e \wedge$ "Greek" $\in l_e^{dbo:FM,dbo:type,so} \Rightarrow (e, dbo{:}type, dbr{:}Prefectures\_of\_Greece) \in G$
0.87	$dbo{:}FM \in c_e \wedge$ "Greek" $\in l_e(c, p, so) \Rightarrow (e, dbp{:}subdivisionName, dbr{:}Greece) \in G$
0.83	$dbo{:}President \in c_e \wedge (e, dbo{:}nationality, dbr{:}Greece) \in G \Rightarrow$ "Greek" $\in l_e^{dbo:President,dbo:nationality,so}$
0.83	$dbo{:}President \in c_e \wedge$ "Greek" $\in l_e^{dbo:President,dbo:nationality,so} \Rightarrow (e, dbo{:}nationality, dbr{:}Greece) \in G$
0.82	$dbo{:}Swimmer \in c_e \wedge (e, dbo{:}birthPlace, dbr{:}Greece) \in G \Rightarrow$ "Greek" $\in l_e^{dbo:Swimmer,dbo:birthPlace,so}$
0.82	$dbo{:}Swimmer \in c_e \wedge$ "Greek" $\in l_e^{dbo:Swimmer,dbo:birthPlace,so} \Rightarrow (e, dbo{:}birthPlace, dbr{:}Greece) \in G$
0.82	$dbo{:}Model \in c_e \wedge (e, dbp{:}birthPlace, dbr{:}Greece) \in G \Rightarrow$ "Greek" $\in l_e^{dbo:Model,dbp:birthPlace,os}$
0.82	$dbo{:}RugbyClub \in c_e \wedge (e, dbp{:}location, dbr{:}Greece) \in G \Rightarrow$ "Greek" $\in l_e^{dbo:RugbyClub,dbp:location,so}$
0.82	$dbo{:}Model \in c_e \wedge$ "Greek" $\in l_e^{dbo:Model,dbp:birthPlace,os} \Rightarrow (e, dbp{:}birthPlace, dbr{:}Greece) \in G$
0.82	$dbo{:}RugbyClub \in c_e \wedge$ "Greek" $\in l_e^{dbo:RugbyClub,dbp:location,so} \Rightarrow (e, dbp{:}location, dbr{:}Greece) \in G$

QALD (Question Answering over Linked Data)[5] challenge series[6] that consists of 601 pairs. For each (question, query) pair (t,q), we tokenize t and create a set of linguistic patterns in the same way as we have processed the abstracts and extracted patterns, explained in Section 3. For each query q we create the (possibly empty) sets s_q, p_q, o_q, sp_q, and po_q that are defined as follows. s_q is the set of terms that occur in subject position of triple patterns in q, p_q is the set of terms that occur in predicate position of triple patterns in q, o_q is the set of terms that occur in object position of triple patterns in q, sp_q is a set of tuples of the form (t_1, t_2) where q contains a triple pattern with t_1 in subject position and t_2 in predicate position, and po_q is a set of tuples of the form (t_1, t_2) where q contains a triple pattern with t_1 in predicate position and t_2 in object position. From the set p_q we removed the term *rdfs:label* and the term *rdf:type*, and from the sets sp_q and po_q we removed all pairs of terms that contained the term *rdfs:label* or the term *rdf:type*, because in the experiment we decided against learning rules that are class-specific and that mention another type or that predict a label, although this might be included in the future. q_s was non-empty for 315 queries, q_p was non-empty for 579 queries, q_o was non-empty for 322 queries, q_{sp} was non-empty for 311 queries, and q_{po} was non-empty for 229 queries. 275 distinct terms occurred in subject position, 298 distinct terms occurred in predicate position, 296 distinct terms occurred in object position, 309 distinct term pairs occurred in subject-predicate position, and 259 distinct term pairs occurred in predicate-object position.

As an example, consider the following SPARQL query which corresponds to the question *Give me English actors starring in Lovesick.*[7]

```
SELECT DISTINCT ?uri WHERE {
    res:Lovesick dbo:starring ?uri .
    { ?uri dbo:birthPlace res:England . }
    UNION
    { ?uri rdf:type yago:EnglishFilmActors . }
}
```

Given the SPARQL query above the sets have the following content: $s_q = \{res{:}Lovesick\}$, $p_q = \{dbo{:}starring, dbo{:}birthPlace\}$, $o_q = \{res{:}England, yago{:}EnglishFilmActors\}$, $sp_q = \{(res{:}Lovesick, dbo{:}starring)\}$, $po_q = \{(dbo{:}birthPlace, res{:}England)\}$. The set of linguistic patterns l_q contains the 1-grams *"actors"*, *"Give"*, *"English"*, *"Lovesick"*, and *"starring"*, the 2-grams *"Give me"*, *"actors starring"*, *"me English"*, *"in Lovesick"*, *"starring in"*, and *"English actors"*, and so forth up to 5-grams.

Given a (question, query) pair, we can now find all rules for the 10 rule patterns $c_s, l_s \Rightarrow po$; $c_s, ll_s \Rightarrow po$; $c_s, l_s \Rightarrow p$; $c_s, ll_s \Rightarrow p$; $c_s, l_s \Rightarrow o$; $c_o, l_o \Rightarrow sp$; $c_o, ll_o \Rightarrow sp$; $c_o, l_o \Rightarrow s$; $c_o, l_o \Rightarrow p$; and $c_o, ll_o \Rightarrow p$, i.e., those that predict KB terms based on linguistic patterns. For all these rule patterns, a triple pattern occurs on the right side of the association rules. For a rule r, s_r denotes the triple pattern's subject term, p_r denotes the triple pattern's predicate term, and o_r denotes the triple pattern's object term.

[5] See `http://qald.aksw.org/`

[6] We used all files containing (question, query) pairs from the QALD challenge series that we could get hold on. We used the files `dbpedia-test.xml` and `dbpedia-train.xml` from QALD-1, QALD-2, and QALD-3, the files `qald-4_multilingual_test.xml` and `qald-4_multilingual_train.xml` from QALD-4, the file `qald-5_train.xml` from QALD-5, the files `qald-6-test-multilingual.json` and `qald-6-train-multilingual.json` from QALD-6, the file `qald-7-train-multilingual.json` from QALD-7, and the file `qald-9-train-multilingual.json`. In the case where a question appeared in several challenges we only make use of the corresponding query from the most recent challenge.

[7] The example is taken from the QALD-5 challenge, question #293, file *qald-5_train.xml*.

Let R be a set of rules and let Q be a set of (question, query) pairs. Given a set of rules R and a query q, the set of true positives for predicate terms, denoted by $TP_p(q, R)$, is the set of terms that are necessary for building the query (i.e., those terms that exist in predicate position in the query) and that are proposed by some rule $r \in R$. Likewise, we can define TP_s, TP_o, TP_{sp}, and TP_{po}. The set $FP_p(q, R)$ of false positives for predicate terms is the set of terms that are incorrectly proposed as necessary for building the query (i.e., those terms that exist in predicate position in the query) and that are proposed by some rule $r \in R$. Likewise, we can define FP_s, FP_o, FP_{sp}, and FP_{po}. The set $FN_p(q, R)$ of false negatives for predicate terms is the set of terms that are necessary for building the query (i.e., those terms that exist in predicate position in the query) but are not proposed by any rule $r \in R$. Likewise, we can define FN_s, FN_o, FN_{sp}, and FN_{po}. Given TP_x, FP_x, and FN_x, we can calculate micro-averaged precision ($micro{-}P_x(Q, R)$), micro-averaged recall ($micro{-}R_x(Q, R)$), micro-averaged $F1$ ($micro{-}F1_x(Q, R)$), macro-averaged precision ($macro{-}P_x(Q, R)$), macro-averaged recall ($macro{-}R_x(Q, R)$), and macro-averaged $F1$ ($macro{-}F1_x(Q, R)$) for each prediction type $x \in \{s, p, o, sp, po\}$.

Within the set of non-localized rules we found 17,165,819[8] rules that contain a linguistic pattern that appears in a QALD question. In the set of localized rules we found 742,891[9] rules that contain a linguistic pattern that appears in a QALD question. From these rules, only for 128,223 (≈1%) non-localized rules and for 42,838 (≈6%) localized rules there exists a (question, query) pair such that the rule contains a linguistic pattern that exists in the question and the rule predicts a term or a pair of terms that occurs in the query – thus, these are the desired/helpful rules.[10]

Without filtering the set R of rules, we measured the recall values, because these help us to understand the upper bounds for recall for any subset of R. For the set of non-localized (localized) rules, we measured the following values: $micro{-}R_s$=0.08 (0.03), $microR_p$=0.92 (0.74), $micro{-}R_o$=0.31 (0.21), $micro{-}R_{sp}$=0.02 (0.01), $micro{-}R_{po}$=0.47 (0.3), $macro{-}R_s$=0.08 (0.02), $macro{-}R_p$=0.92 (0.71), $macroR_o$=0.27 (0.16), $macro{-}R_{sp}$=0.02 (0), and $macro{-}R_{po}$ =0.44 (0.26). All precision values were close to zero. It can be seen that the localized rules do not perform better than the non-localized rules.

We investigated the impact of the individual parameters on precision, recall and $F1$ for non-localized and for localized rules. We filtered R with $sup(A), sup(B) \in \{5, 10, 15, 20\}$, $sup(AB) \in \{5, 10, 15\}$, $P(B|A), P(A|B) \in \{0, 0.01, 0.02, 0.03, 0.04, 0.05\}$, and for the *AllConf* measure the threshold values $\{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$. Instead of exploring the cartesian product of possible parameter value combinations, for each experiment we only let one parameter take a value that is not the lowest possible value, which results in a set of 28 experiments. For localized rules, Figure 1 shows the precision values for each experiment, Figure 2 shows the recall values for each experiment, and Figure 3 shows the $F1$ values

[8] $c_s, l_s \Rightarrow o$: 5,599,910, $c_o, l_o \Rightarrow p$: 529,331, $c_s, l_s \Rightarrow p$: 3,828,243, $c_s, l_s \Rightarrow po$: 6,395,776, $c_o, l_o \Rightarrow s$: 59,584, $c_o, l_o \Rightarrow sp$: 752,974

[9] $c_o, ll_o \Rightarrow sp$: 41,204, $c_o, ll_o \Rightarrow p$: 30,870, $c_s, ll_s \Rightarrow po$: 487,176, $c_s, ll_s \Rightarrow p$: 409,408

[10] Objects were correctly predicted by 8,044 rules of type $c_s, l_s \Rightarrow o$, 16,207 rules of type $c_s, l_s \Rightarrow po$, and 6,625 rules of type $c_s, ll_s \Rightarrow po$; predicates were correctly predicted by 5,005 rules of type $c_o, l_o \Rightarrow p$, 25,127 rules of type $c_s, l_s \Rightarrow p$, 107,186 rules of type $c_s, l_s \Rightarrow po$, 15,626 rules of type $c_o, l_o \Rightarrow sp$, 1,384 rules of type $c_o, ll_o \Rightarrow p$, 10,392 rules of type $c_s, ll_s \Rightarrow p$, 24,709 rules of type $c_s, ll_s \Rightarrow po$, and 1,571 rules of type $c_o, ll_o \Rightarrow sp$; subjects were correctly predicted by 16 rules of type $c_o, l_o \Rightarrow s$, 100 rules of type $c_o, l_o \Rightarrow sp$, and 37 rules of type $c_o, ll_o \Rightarrow sp$; subject-predicate pairs were correctly predicted by 9 rules of type $c_o, l_o \Rightarrow sp$ and 2 rules of type $c_o, ll_o \Rightarrow sp$; property-object pairs were correctly predicted by 3,173 rules of type $c_s, l_s \Rightarrow po$ and by 1,577 rules of type $c_s, ll_s \Rightarrow po$.

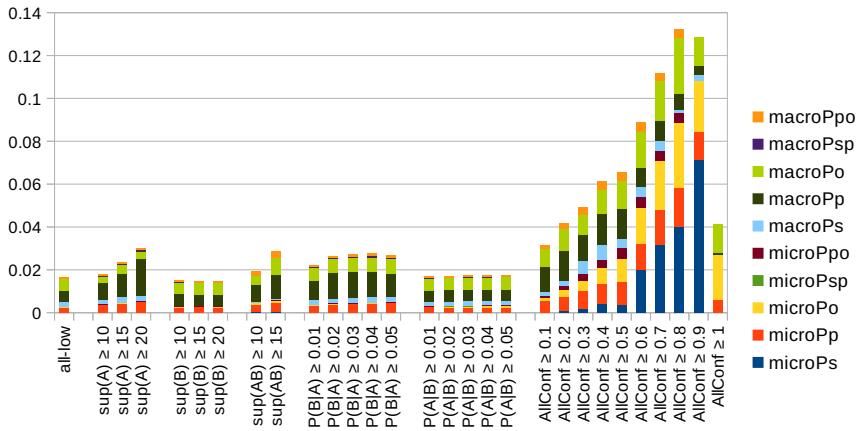


Figure 1 Precision values for each of the 28 experiments with localized rules.

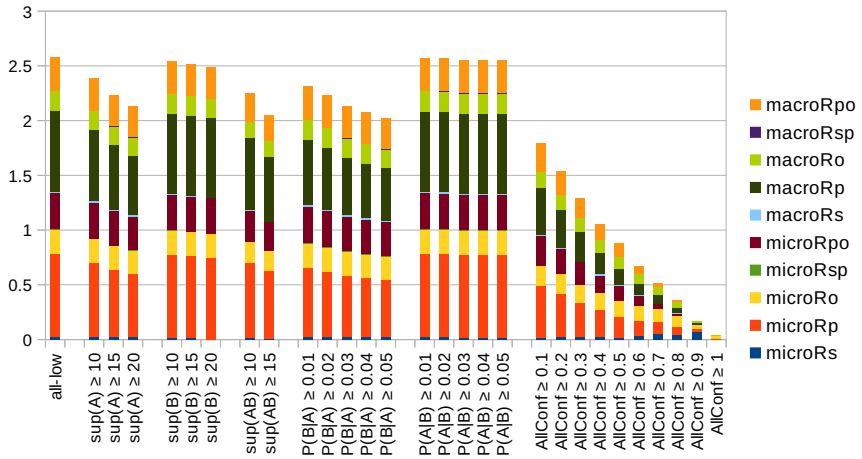


Figure 2 Recall values for each of the 28 experiments with localized rules.

for each experiment. The interestingness threshold appears to have the highest impact on precision, recall, and $F1$. However, increasing the threshold for the *AllConf* measure also decreases precision. Note that due to the bar chart being stacked, recall values can be above 1, because each recall value, e.g., $macro{-}R_{po}$, is a value in the range $[0, 1]$.

4.1 Gold Standard Evaluation

The evaluation described previously considers all possible pairs of lexical elements and KB elements that can be extracted from pairs of NL question and SPARQL queries in the QALD dataset. In order to allow for a more controlled evaluation that allows us to examine the performance of our approach on different parts-of-speech, we manually created a gold standard from QALD-9 for three parts-of-speech: for adjectives referring to a pair of property and object, for verbs referring to a property, and for (relational) nouns referring to a property. We describe the gold standards for the three different parts-of-speech in the following:

- Gold standard for adjectives: comprising of 13 adjectives referring to a pair of property and object. As an example, the adjective *Swedish* in the question *"Give me all Swedish holidays"* refers to the pair $(dbo{:}country, res{:}Sweden)$.

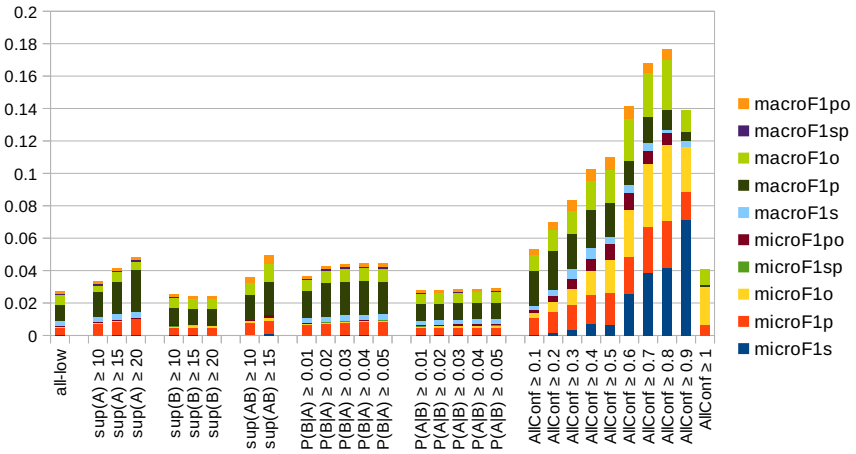


Figure 3 $F1$ values for each of the 28 experiments with localized rules.

- Gold standard for verbs: comprising of 69 verbs referring to a property. As an example, the verb *dissolve* in the question *"When did the Ming dynasty dissolve?"* refers to the property *dbo*:*dissolutionDate*.
- Gold standard for (relational) nouns: comprising of 55 nouns. As an example, the relational noun *founder (of)* refers to the property *dbo*:$founder$ in the question *"Who is the founder of Penguin Books?"*.

In Table 4, we give the results in terms of four metrics: MRR, Hits@1, Hits@5, Hits@10. Mean reciprocal rank (MRR) is a measure used in information retrieval to evaluate ranked lists of results. The MRR is defined as follows:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{|rank_i|}$$

In our case the query is the lexical element in question and the retrieved list corresponds to the KB elements ranked by the corresponding interestingness measure. Hits@k denotes the percentage of queries for which the correct KB element is within the top k results. We provide the results for the best configuration in terms of hyperparameters for each part-of-speech.

The best results were obtained for adjectives when we filtered rules that do not satisfy the following constraints: $supA \geq 5$, $supB \geq 50$, $P(A|B) \geq 0.1$, $P(B|A) \geq 0.05$, and an interestingness value ≥ 0.2 Among the interestingness measures, $MaxConf$ achieves higher performance (0.23, 0.35, 0.35, and 0.4 for MRR, Hits@1 and Hits@5, Hits@10 respectively) than all other interestingness measures. The low results in terms of MRR are due to the fact that in some cases, the correct (property, object) pair for an adjective is ranked rather low in the list. For the adjective *Canadian*, the correct pair (*dbo*:*country*, *dbr*:*Canada*) ranks at position 17 of the best ranking with the $MaxConf$ measure, while other related (property, object) pairs that are more specific rank higher, such as (*dbo*:*region*, *dbr*:*Saskatchewan*), (*dbo*:*location*, *dbr*:*Ontario*) etc. The best results were obtained for verbs with the configuration $supA \geq 50$, $supB \geq 50$, $P(A|B) \geq 0.1$, $P(B|A) \geq 0.1$, MaxConf ≥ 0.2. For the majority of verbs including *create*, *design*, *develop*, *die*, *direct*, *found*, *marry*, etc. the corresponding correct property *dbo*:*creator*, *dbo*:*designer*, *dbo*:*developer*, *dbo*:*deathPlace*, *dbo*:*director*, *dbo*:$founder$, *dbo*:*spouse* rank at position 1. The best results were obtained for relative nouns with the configuration $supA \geq 50$, $supB \geq 50$, $P(A|B) \geq 0.1$, $P(B|A) \geq 0.05$, IR ≥ 0.2.

Table 4 Results of Gold Standard Evaluation of three parts-of-speech: adjective, verb, and noun.

Measure	Adjective				Verb				Noun			
	MRR	Hits1	Hits5	Hits10	MRR	Hits1	Hits5	Hits10	MRR	Hits1	Hits5	Hits10
Cosine	0.08	0.05	0.2	0.2	0.28	0.25	0.31	0.31	0.19	0.18	0.2	0.2
Coherence	0.05	0.0	0.15	0.2	0.28	0.25	0.31	0.31	0.19	0.18	0.2	0.2
AllConf	0.04	0.0	0.15	0.2	0.28	0.25	0.31	0.31	0.19	0.18	0.2	0.2
MaxConf	**0.23**	**0.35**	**0.35**	**0.4**	**0.31**	**0.31**	**0.31**	**0.31**	0.19	0.18	0.2	0.2
IR	0.13	0.05	0.2	0.4	0.31	0.31	0.31	0.31	**0.2**	**0.2**	**0.2**	**0.2**
Kulczynski	0.11	0.05	0.2	0.3	0.28	0.25	0.31	0.31	0.19	0.18	0.2	0.2

5 Related Work

Related work can be grouped into two areas: i) (mining of patterns for) information extraction from text to RDF, and ii) (mining of patterns for) natural language generation from RDF.

Several works, such as by Gerber et al. [7], Nakashole et al. [13], and Walter et al. [15], apply the distant supervision principle to extract relation-specific patterns from natural language sentences. In our framework, relation-specific patterns can be expressed with the association rule patterns $c_s, ll_s \Rightarrow p$ and $c_o, ll_o \Rightarrow p$.

Gerber et al. [7] apply their approach to texts from Wikipedia and DBpedia as KB. The patterns, called BOA patterns, can be used to extract relations from texts and to populate a knowledge base with the extraction results. BOA patterns are scored based on support, as we propose as well, but furthermore BOA patterns are scored on typicity and specificity, whereas we make use of conditional probabilities and interestingness measures. An example of a BOA pattern for the predicate *subsidiary* is *?D?'s acquisition of ?R?*. Here, *?D?* and *?R?* matches entities that are instances of the classes specified as the domain and range of the predicate, respectively. Thus, a BOA pattern can be specific to up to two classes.

Nakashole et al. [13] introduce SOL patterns. These patterns can consist of syntactic features, ontological type signatures, and lexical features. In contrast to our approach, the authors extract patterns from dependency-parsed sentences instead of from tokenized texts and collect dependency paths between identified entities. Patterns are scored by support and confidence. An example of an SOL pattern for the relation *hasMusicalIdol* is *<musician> PRP idol <musician>*, where *musician*, the ontological type signature, matches any entity that is an instance of the class *musician* and *PRP* matches any token that is a pronoun.

The approach M-ATOLL by Walter et al. [15] mines textual patterns that denote binary relations between entities. The text corpus is dependency-parsed and natural language patterns are identified via a set of manually defined dependency graph patterns that are matched against the parsed text. The resulting patterns are represented in *lemon* [9] format. Going beyond M-ATOLL, we do not rely on a pre-definend set of patterns, but mine the patterns inductively from data (that has not been dependency-parsed).

In contrast to the previous three approaches, although also extracting relations from Wikipedia abstracts and making use of the distant supervision principle, Heist et al. [8] propose an approach that does not make use of linguistic features, for example by considering the position of an identified entity in an abstract. The authors train several classification algorithms and show that a classifier trained on one language can also classify relations in another language, which is possible since the features aren't language-specific in the sense that they do not make use of lexical or syntactic information.

Ding et al. [4] propose an approach to map adjectives to existential restrictions over a KB. Their approach, Adj2ER, finds for example that the adjective *American* can be expressed via the existential restriction $\exists dbo{:}nationality.\{dbr{:}United\_States\}$. This existential restriction

is comparable to the rule pattern $c_s, l_s \Rightarrow po$. As a further similarity, the authors take into account which class an adjective modifies. Adj2ER can create existential restrictions that contain negations. For example, the approach finds that for instances of the class *Actor* the adjective *alive* can be mapped to $\neg\exists deathDate.\top$. Negation cannot be expressed within our framework of association rules. Instead of distant supervision on natural language text, for an adjective and a class their approach collects entities that are instances of that class and then create two sets: one set where the instance and the adjective co-occur in some text and the other set of entities that do not. Then, they make use of the information in a KB about these entities to derive the existential restrictions.

A simple form of generation of natural language text from RDF can be realized, as Sun and Mellish [14] show, by categorizing the names of terms such as predicates (e.g., "has" + noun) and by making use of a few templates specific to these categories. The approach requires the names in an ontology to follow certain conventions and creates verbalizations that may not always be natural. Moreover, each triple is verbalized as an individual sentence. A possibility to create verbalizations that are natural in style is to make use of a lexicon, as shown by Cimiano et al. [3]. However, such a lexicon may not always be available. Ell and Harth [5] present an approach that applies the distant supervision principle and automatically extracts verbalization templates that express multiple triples in one sentence. A good overview about NLG from RDF can be found in the context of the WebNLG challenge[11] [6]. Approaches that tackle this challenge need to be able to carry out tasks such as sentence segmentation, lexicalization, aggregation, and surface realisation. Those association rules mined by our approach that predict a linguistic pattern could be applicable in the context of the lexicalization task. Recent work by Moussallem et al. [11] presents an approach based on a encoder-decoder architecture that is capable of generating multilingual verbalizations.

6 Conclusion

We have presented *LexExMachina*, a new approach to closing the gap between lexicon and ontology by mining a set of 20 types of class-specific association rules that connect a lexical element to a data element from a KB. These rules can be used for information extraction, question answering as well as KB verbalization tasks. We have mined association rules from the loosely-parallel corpus consisting of Wikipedia and DBpedia for the 354 classes that have at least 100 instances. The resulting rules have been evaluated on a QA task of reconstructing all the elements of the query from the NL question by relying on these correspondences as well as on a manually created gold standard that allows us to inspect the results for different parts-of-speech. Our framework subsumes many of the pattern mining approaches proposed so far and shows promising results. Although our experiment showed that high-quality and high-coverage association rules can be found, for example those that contain the token *Greek*, shown in Table 3, we need to investigate further how to increase precision without severely sacrificing recall. Beyond the seven parameters taken into account so far, we plan to investigate the impact of further parameters, such as the length of a string between two arguments from which the patterns are extracted, and how fuzzy matching can help to increase recall.

[11] See `https://webnlg-challenge.loria.fr/`

References

1 Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. DBpedia: A Nucleus for a Web of Open Data. In *The semantic web*, pages 722–735. Springer, 2007.

2 Sergey Brin, Rajeev Motwani, and Craig Silverstein. Beyond Market Baskets: Generalizing Association Rules to Correlations. In *Proceedings of the 1997 ACM SIGMOD international conference on Management of data*, pages 265–276, 1997.

3 Philipp Cimiano, Janna Lüker, David Nagel, and Christina Unger. Exploiting ontology lexica for generating natural language texts from RDF data. In *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 10–19, 2013.

4 Jiwei Ding, Wei Hu, Qixin Xu, and Yuzhong Qu. Mapping Factoid Adjective Constraints to Existential Restrictions over Knowledge Bases. In *ISWC*, pages 164–181. Springer, 2019.

5 Basil Ell and Andreas Harth. A language-independent method for the extraction of rdf verbalization templates. In *Proceedings of the 8th international natural language generation conference (INLG)*, pages 26–34, 2014.

6 Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. The WebNLG challenge: Generating text from RDF data. In *INLG*, pages 124–133, 2017.

7 Daniel Gerber and A-C Ngonga Ngomo. Bootstrapping the Linked Data Web. In *1st Workshop on Web Scale Knowledge Extraction@ ISWC*, volume 2011, 2011.

8 Nicolas Heist and Heiko Paulheim. Language-Agnostic Relation Extraction from Wikipedia Abstracts. In *The Semantic Web – ISWC 2017*, pages 383–399, 2017.

9 John McCrae, Dennis Spohr, and Philipp Cimiano. Linking Lexical Resources and Ontologies on the Semantic Web with Lemon. In *ESWC*, pages 245–259. Springer, 2011.

10 Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. Distant supervision for relation extraction without labeled data. In *ACL-IJCNLP 2009*, pages 1003–1011, 2009.

11 Diego Moussallem, Dwaraknath Gnaneshwar, Thiago Castro Ferreira, and Axel-Cyrille Ngonga Ngomo. NABU–Multilingual Graph-Based Neural RDF Verbalizer. In *ISWC*, pages 420–437, 2020.

12 Diego Moussallem, René Speck, and Axel-Cyrille Ngonga Ngomo. Generating Explanations in Natural Language from Knowledge Graphs. In *Knowledge Graphs for eXplainable Artificial Intelligence: Foundations, Applications and Challenges*, volume 47 of *Studies on the Semantic Web*, pages 213–241. IOS Press, 2020.

13 Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. PATTY: A Taxonomy of Relational Patterns with Semantic Types. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1135–1145, 2012.

14 Xiantang Sun and Chris Mellish. An experiment on "free generation" from single rdf triples. In *Proceedings of the Eleventh European Workshop on Natural Language Generation (ENLG 07)*, pages 105–108, 2007.

15 Sebastian Walter, Christina Unger, and Philipp Cimiano. M-ATOLL: A Framework for the Lexicalization of Ontologies in Multiple Languages. In *ISWC*, pages 472–486. Springer, 2014.

16 Tianyi Wu, Yuguo Chen, and Jiawei Han. Re-examination of interestingness measures in pattern mining: a unified framework. *Data Mining and Knowledge Discovery*, 21(3):371–397, 2010.

A Details on and Examples for the Rule Patterns

Rule Patterns $c_s, l_s \Rightarrow po$ and $c_s, ll_s \Rightarrow po$. Given for a rule of type $c_s, l_s \Rightarrow po$ ($c_s, ll_s \Rightarrow po$) are a class $c \in C$, a property $p \in P$, a term $o \in \mathcal{T}$, and a (localized) linguistic pattern l.

$$c \in c_e \wedge l \in l_e \Rightarrow (e, p, o) \in G \qquad (c_s, l_s \Rightarrow po)$$
$$c \in c_e \wedge l \in l_e^{c,p,d} \Rightarrow (e, p, o) \in G \qquad (c_s, ll_s \Rightarrow po)$$

Meaning: Given that in a document that describes an entity e that is an instance of the class c occurs the (localized) linguistic pattern l, the rule predicts that the entity e has the value o for the property p.

Example for rule pattern $c_s, l_s \Rightarrow po$:

$dbo{:}Politician \in c_e \wedge$ "Awami League" $\in l_e$
$\Rightarrow (e, dbo{:}party, dbr{:}Bangladesh\_Awami\_League) \in G$

sup(A)= 40	AllConf(A,B)≈ 0.88
sup(B)= 42	Coherence(A,B)≈ 0.45
sup(AB)= 37	Cosine(A,B)≈ 0.9
P(B|A)≈ 0.88	IR(A,B)≈ 0.04
P(A|B)≈ 0.92	Kulczynski(A,B)≈ 0.9
	MaxConf(A,B)≈ 0.92

Meaning: Given an entity that is an instance of the class $dbo{:}Politician$ and where the document that describes that entity contains the linguistic pattern "Awami League", the rule predicts that the entity is in the relation $dbo{:}party$ with $dbr{:}Bangladesh\_Awami\_League$.

Example for rule pattern $c_s, ll_s \Rightarrow po$:

$dbo{:}Arachnid \in c_e \wedge$
"family Trombidiidae" $\in l_e^{dbo:Arachnid,dbo:genus,so}$
$\Rightarrow (e, dbo{:}genus, dbr{:}Trombidium) \in G$

sup(A)= 40	AllConf(A,B)≈ 0.88
sup(B)= 42	Coherence(A,B)≈ 0.45
sup(AB)= 37	Cosine(A,B)≈ 0.9
P(B|A)≈ 0.88	IR(A,B)≈ 0.04
P(A|B)≈ 0.92	Kulczynski(A,B)≈ 0.9
	MaxConf(A,B)≈ 0.92

Meaning: Given an entity that is an instance of the class $dbo{:}Arachnid$ and where the abstract of that entity contains the localized linguistic pattern "family Trombidiidae" (which is localized to the class $dbo{:}Arachnid$ and the predicate $dbo{:}genus$), the rule predicts that the entity is in the relation $dbo{:}genus$ with $dbr{:}Trombidium$.

Rule Patterns $c_s, l_s \Rightarrow p$ and $c_s, ll_s \Rightarrow p$. Given for a rule of type $c_s, l_s \Rightarrow p$ ($c_s, ll_s \Rightarrow p$) are a class $c \in C$, a property $p \in P$, and a (localized) linguistic pattern l.

$$c \in c_e \wedge l \in l_e \Rightarrow \exists o \in \mathcal{T} : (e, p, o) \in G \qquad (c_s, l_s \Rightarrow p)$$
$$c \in c_e \wedge l \in l_e^{c,p,d} \Rightarrow \exists o \in \mathcal{T} : (e, p, o) \in G \qquad (c_s, ll_s \Rightarrow p)$$

Meaning: Given that in a document that describes an entity e that is an instance of the class c occurs the linguistic pattern l, predict that the entity e has some value for the property p.

Example for rule pattern $c_s, l_s \Rightarrow p$:

$dbo{:}Actor \in c_e \wedge$ "a Swedish actor" $\in l_e$
$\Rightarrow \exists o : (e, dbo{:}nationality, o) \in G$

sup(A)= 213	AllConf(A,B)≈ 0.29
sup(B)= 729	Coherence(A,B)≈ 0.23
sup(AB)= 213	Cosine(A,B)≈ 0.54
P(B|A)≈ 0.29	IR(A,B)≈ 0.71
P(A|B)≈ 1	Kulczynski(A,B)≈ 0.65
	MaxConf(A,B)≈ 1

Meaning: Given an entity that is an instance of the class $dbo{:}Actor$ and where the document that describes that entity contains the linguistic pattern "a Swedish actor", the rule predicts that the entity is in the relation $dbo{:}nationality$ with some entity.

Example for rule pattern $c_s, ll_s \Rightarrow p$:

$dbo{:}Actor \in c_e \wedge$ "married to" $\in l_e^{dbo:Actor,dbo:spouse,so}$	sup(A)= 61 AllConf(A,B)≈ 0.12
$\Rightarrow \exists o : (e, dbo{:}spouse, o) \in G$	sup(B)= 289 Coherence(A,B)≈ 0.1
	sup(AB)= 36 Cosine(A,B)≈ 0.27
	P(B|A)≈ 0.12 IR(A,B)≈ 0.73
	P(A|B)≈ 0.59 Kulczynski(A,B)≈ 0.36
	MaxConf(A,B)≈ 0.59

Meaning: Given an entity that is an instance of the class *dbo*:*Actor* and where the document that describes that entity contains the linguistic pattern "married to" (which is localized to the class *dbo*:*Actor* and the predicate *dbo*:*spouse*), the rule predicts that the entity is in the relation *dbo*:*spouse* with some entity.

Rule Pattern $c_s, l_s \Rightarrow o$. Given for a rule of type $c_s, l_s \Rightarrow o$ are a class $c \in C$, a term $o \in \mathcal{T}$, and a linguistic pattern l.

$$c \in c_e \wedge l \in l_e \Rightarrow \exists p \in \mathcal{U} : (e, p, o) \in G \qquad (c_s, l_s \Rightarrow o)$$

Meaning: Given that in a document that describes an entity e that is an instance of the class c occurs the linguistic pattern l, predict that there is some relation by which e is related to the term o.

Example for rule pattern $c_s, l_s \Rightarrow o$:

$dbo{:}Grape \in c_e \wedge$ "white" $\in l_e$	sup(A)= 225 AllConf(A,B)≈ 0.81
$\Rightarrow \exists p : (e, p,$ "Blanc"@$en) \in$	sup(B)= 198 Coherence(A,B)≈ 0.43
	sup(AB)= 183 Cosine(A,B)≈ 0.87
	P(B|A)≈ 0.92 IR(A,B)≈ 0.11
	P(A|B)≈ 0.81 Kulczynski(A,B)≈ 0.87
	MaxConf(A,B)≈ 0.92

Meaning: Given an entity that is an instance of the class *dbo*:*Grape* and where the document that describes that entity contains the linguistic pattern "white", the rule predicts that the entity is in some relation with the term "Blanc"@*en*.

Rule Patterns $c_o, l_o \Rightarrow sp$ **and** $c_o, ll_o \Rightarrow sp$. Given for a rule of type $c_o, l_o \Rightarrow sp$ ($c_o, ll_o \Rightarrow sp$) are a class $c \in C$, a term $s \in \mathcal{U} \cup \mathcal{B}$, a predicate $p \in \mathcal{P}$, and a (localized) linguistic pattern l.

$$c \in c_e \wedge l \in l_e \Rightarrow (s, p, e) \in G \qquad (c_o, l_o \Rightarrow sp)$$
$$c \in c_e \wedge l \in l_e^{c,p,d} \Rightarrow (s, p, e) \in G \qquad (c_o, ll_o \Rightarrow sp)$$

Meaning: Given that in a document that describes an entity e that is an instance of the class c occurs the (localized) linguistic pattern l, predict that there is an entity s that is in relation p with the entity e.

Example for rule pattern $c_o, l_o \Rightarrow sp$:

$dbo{:}Island \in c_e \wedge$ "Baltic" $\in l_e$	sup(A)= 43 AllConf(A,B)≈ 0.35
$\Rightarrow (dbr{:}Baltic\_Sea, dbo{:}island, e) \in G$	sup(B)= 23 Coherence(A,B)≈ 0.23
	sup(AB)= 15 Cosine(A,B)≈ 0.48
	P(B|A)≈ 0.65 IR(A,B)≈ 0.39
	P(A|B)≈ 0.35 Kulczynski(A,B)≈ 0.5
	MaxConf(A,B)≈ 0.65

Meaning: Given an entity that is an instance of the class *dbo*:*Island* and where the document that describes that entity contains the linguistic pattern "Baltic", the rule predicts that the entity *dbr*:*Baltic_See* is in the relation *dbo*:*island* with this entity.

Example for rule pattern $c_o, ll_o \Rightarrow sp$:

$dbo{:}Artwork \in c_e \wedge$ "Salvador" $\in l_e^{dbo:Artwork,dbo:notableWork,so}$	sup(A)= 6 AllConf(A,B)≈ 0.86
$\Rightarrow (dbr{:}Salvador\_Dalí, dbo{:}notableWork, e) \in G$	sup(B)= 7 Coherence(A,B)≈ 0.46
	sup(AB)= 6 Cosine(A,B)≈ 0.93
	P(B|A)≈ 0.86 IR(A,B)≈ 0.14
	P(A|B)≈ 1 Kulczynski(A,B)≈ 0.93
	MaxConf(A,B)≈ 1

Meaning: Given an entity that is an instance of the class *dbo*:*Artwork* and where the document that describes that entity contains the linguistic pattern "Salvador" (which is localized to the class *dbo*:*Artwork* and the predicate *dbo*:*notableWork*), the rule predicts that the entity *dbr*:*Salvador_Dalí* is in the relation *dbo*:*notableWork* with this entity.

Rule Pattern $c_o, l_o \Rightarrow s$. Given for a rule of type $c_o, l_o \Rightarrow s$ are a class $c \in C$, a term $o \in \mathcal{T}$, and a linguistic pattern l.

$$c \in c_e \wedge l \in l_e \Rightarrow (s, p, e) \in G \qquad (c_o, l_o \Rightarrow s)$$

Meaning: Given that in a document that describes an entity e that is an instance of the class c occurs the linguistic pattern l, predict that there is an entity s that is in some relation with the entity e.

Example for rule pattern $c_o, l_o \Rightarrow s$:

dbo:*Language* $\in c_e \wedge$ "Nahuatl" $\in l_e$	sup(A)= 21	AllConf(A,B)≈ 0.76
	sup(B)= 18	Coherence(A,B)≈ 0.41
$\Rightarrow \exists p : (dbr{:}Nahuan\_languages, p, e) \in G$	sup(AB)= 16	Cosine(A,B)≈ 0.82
	P(B|A)≈ 0.89	IR(A,B)≈ 0.13
	P(A|B)≈ 0.76	Kulczynski(A,B)≈ 0.83
		MaxConf(A,B)≈ 0.89

Meaning: Given an entity that is an instance of the class *dbo*:*Language* and where the document that describes that entity contains the linguistic pattern "Nahuatl", the rule predicts that the entity *dbr*:*Nahuan_languages* is in some relation with this entity.

Rule Patterns $c_o, l_o \Rightarrow p$ **and** $c_o, ll_o \Rightarrow p$. Given for a rule of type $c_o, l_o \Rightarrow p$ ($c_o, ll_o \Rightarrow p$) are a class $c \in C$, a predicate $p \in \mathcal{P}$, and a (localized) linguistic pattern l.

$$c \in c_e \wedge l \in l_e \Rightarrow \exists s \in \mathcal{U} \cup \mathcal{B} : (s, p, e) \in G \qquad (c_o, l_o \Rightarrow p)$$
$$c \in c_e \wedge l \in l_e^{c,p,d} \Rightarrow \exists s \in \mathcal{U} \cup \mathcal{B} : (s, p, e) \in G \qquad (c_o, ll_o \Rightarrow p)$$

Meaning: Given that in a document that describes an entity e that is an instance of the class c occurs the (localized) linguistic pattern l, predict that there is some entity that is in the relation p with the entity e.

Example for rule pattern $c_o, l_o \Rightarrow p$:

dbo:*AmateurBoxer* $\in c_e \wedge$ "silver medal" $\in l_e$	sup(A)= 70	AllConf(A,B)≈ 0.31
	sup(B)= 29	Coherence(A,B)≈ 0.22
$\Rightarrow \exists s : (s, dbo{:}silverMedalist, e) \in G$	sup(AB)= 22	Cosine(A,B)≈ 0.49
	P(B|A)≈ 0.76	IR(A,B)≈ 0.53
	P(A|B)≈ 0.31	Kulczynski(A,B)≈ 0.54
		MaxConf(A,B)≈ 0.76

Meaning: Given an entity e that is an instance of the class *dbo*:*AmateurBoxer* and where the document that describes that entity contains the linguistic pattern "silver medal", the rule predicts that there is some entity which is related via the relation *dbo*:*silverMedalist* to the entity e.

Example for rule pattern $c_o, ll_o \Rightarrow p$:

dbo:*Noble* $\in c_e \wedge$ "married" $\in l_e^{dbo:Noble,dbo:spouse,so}$	sup(A)= 220	AllConf(A,B)≈ 0.1
	sup(B)= 1588	Coherence(A,B)≈ 0.08
$\Rightarrow \exists s : (s, dbo{:}spouse, e) \in G$	sup(AB)= 151	Cosine(A,B)≈ 0.26
	P(B|A)≈ 0.1	IR(A,B)≈ 0.83
	P(A|B)≈ 0.69	Kulczynski(A,B)≈ 0.39
		MaxConf(A,B)≈ 0.69

Meaning: Given an entity e that is an instance of the class *dbo*:*Noble* and where the document that describes that entity contains the linguistic pattern "married" (which is localized to the class *dbo*:*Noble* and the predicate *dbo*:*spouse*), the rule predicts that there is some entity which is related via the relation *dbo*:*spouse* to the entity e.

Rule Patterns $c_s, po \Rightarrow l_s$ and $c_s, po \Rightarrow ll_s$. Given for a rule of type $c_s, po \Rightarrow l_s$ ($c_s, po \Rightarrow ll_s$) are a class $c \in C$, a predicate $p \in \mathcal{P}$, a term $o \in \mathcal{T}$, and a (localized) linguistic pattern l.

$$c \in c_e \wedge (e, p, o) \in G \Rightarrow l \in l_e \qquad (c_s, po \Rightarrow l_s)$$
$$c \in c_e \wedge (e, p, o) \in G \Rightarrow l \in l_e^{c,p,d} \qquad (c_s, po \Rightarrow ll_s)$$

Meaning: Given an entity e that is an instance of the class c and given that e is in relation p to the term o, predict that the text that describes e contains the (localized) linguistic pattern l.

Example for rule pattern $c_s, po \Rightarrow l_s$:

$dbo{:}Actor \in c_e \wedge (e, dbo{:}nationality, dbr{:}Sweden) \in G$	sup(A)= 589	AllConf(A,B)≈ 0.98
$\Rightarrow$ "Swedish" $\in l_e$	sup(B)= 582	Coherence(A,B)≈ 0.49
	sup(AB)= 579	Cosine(A,B)≈ 0.99
	P(B|A)≈ 0.99	IR(A,B)≈ 0.01
	P(A|B)≈ 0.98	Kulczynski(A,B)≈ 0.99
		MaxConf(A,B)≈ 0.99

Meaning: Given an entity e that is an instance of the class *dbo*:*Actor* and where the entity is in the relation *dbo*:*nationality* with the entity *dbr*:*Sweden*, the rule predicts that the linguistic pattern "Swedish" occurs in the text about the entity e.

Example for rule pattern $c_s, po \Rightarrow ll_s$:

$dbo{:}Criminal \in c_e \wedge (e, dbo{:}deathPlace, dbr{:}Sicily) \in G$	sup(A)= 11	AllConf(A,B)≈ 0.64
$\Rightarrow$ "Mafia" $\in l_e^{dbo:Criminal, dbo:deathPlace, os}$	sup(B)= 8	Coherence(A,B)≈ 0.37
	sup(AB)= 7	Cosine(A,B)≈ 0.75
	P(B|A)≈ 0.88	IR(A,B)≈ 0.25
	P(A|B)≈ 0.64	Kulczynski(A,B)≈ 0.76
		MaxConf(A,B)≈ 0.88

Meaning: Given an entity e that is an instance of the class *dbo*:*Criminal* and where the entity is in the relation *dbo*:*deathPlace* with the entity *dbr*:*Sicily*, the rule predicts that the localized linguistic pattern "Mafia" (which is localized to the class *dbo*:*Criminal* and the predicate *dbo*:*deathPlace*) occurs in the text about the entity e.

Rule Patterns $c_s, p \Rightarrow l_s$ and $c_s, p \Rightarrow ll_s$. Given for a rule of type $c_s, p \Rightarrow l_s$ ($c_s, p \Rightarrow l_s$) are a class $c \in C$, a predicate $p \in \mathcal{P}$, and a (localized) linguistic pattern l.

$$c \in c_e \wedge \exists o \in \mathcal{T} : (e, p, o) \in G \Rightarrow l \in l_e \qquad (c_s, p \Rightarrow l_s)$$
$$c \in c_e \wedge \exists o \in \mathcal{T} : (e, p, o) \in G \Rightarrow l \in l_e^{c,p,d} \qquad (c_s, p \Rightarrow ll_s)$$

Meaning: Given an entity e that is an instance of the class c and given that e is in relation p to some term, predict that the text that describes e contains the (localized) linguistic pattern l.

Example for rule pattern $c_s, p \Rightarrow l_s$:

$dbo{:}Fungus \in c_e \wedge \exists o : (e, dbp{:}genusAuthority, o) \in G$	sup(A)= 4330	AllConf(A,B)≈ 0.86
$\Rightarrow$ "is a genus" $\in l_e$	sup(B)= 3773	Coherence(A,B)≈ 0.46
	sup(AB)= 3717	Cosine(A,B)≈ 0.92
	P(B|A)≈ 0.99	IR(A,B)≈ 0.13
	P(A|B)≈ 0.86	Kulczynski(A,B)≈ 0.92
		MaxConf(A,B)≈ 0.99

Meaning: Given an entity e that is an instance of the class *dbo*:*Fungus* and where the entity is in the relation *dbo*:*genusAuthority* with some term, the rule predicts that the linguistic pattern "is a genus" occurs in the text about the entity e.

Example for rule pattern $c_s, p \Rightarrow ll_s$:

$dbo{:}CricketGround \in c_e \wedge \exists o : (e, dbp{:}location, o) \in G$	sup(A)= 195	AllConf(A,B)≈ 0.33
$\Rightarrow$ "is a cricket ground in" $\in l_e^{dbo:CricketGround, dbp:location, so}$	sup(B)= 64	Coherence(A,B)≈ 0.25
	sup(AB)= 64	Cosine(A,B)≈ 0.57
	P(B|A)≈ 1	IR(A,B)≈ 0.67
	P(A|B)≈ 0.33	Kulczynski(A,B)≈ 0.66
		MaxConf(A,B)≈ 1

Meaning: Given an entity e that is an instance of the class *dbo*:*CricketGround* and where the entity is in the relation *dbp*:*location* with some term, the rule predicts that the localized linguistic pattern "is a cricket ground in" (which is localized to the class *dbo*:*CricketGround* and the predicate *dbp*:*location*) occurs in the text about the entity e.

Rule Pattern $c_s, o \Rightarrow l_s$. Given for a rule of type $c_s, o \Rightarrow l_s$ are a class $c \in C$, a term $o \in \mathcal{T}$, and a linguistic pattern l.

$$c \in c_e \wedge \exists p \in \mathcal{U} : (e, p, o) \in G \Rightarrow l \in l_e \qquad (c_s, o \Rightarrow l_s)$$

Meaning: Given an entity e that is an instance of the class c and given that e is in some relation to the term o, predict that the text that describes e contains the (localized) linguistic pattern l.

Example for rule pattern $c_s, o \Rightarrow l_s$:

$dbo{:}Protein \in c_e \wedge \exists p : (e, p, "MT") \in G$	sup(A)= 24	AllConf(A,B)≈ 1
$\Rightarrow$ "Mitochondrially encoded" $\in l_e$	sup(B)= 24	Coherence(A,B)≈ 0.5
	sup(AB)= 24	Cosine(A,B)≈ 1
	P(B|A)≈ 1	IR(A,B)≈ 0
	P(A|B)≈ 1	Kulczynski(A,B)≈ 1
		MaxConf(A,B)≈ 1

Meaning: Given an entity e that is an instance of the class *dbo*:*Protein* and where the entity is in some relation with the term "*MT*", the rule predicts that the linguistic pattern "Mitochondrially encoded" occurs in the text about the entity e.

Rule Patterns $c_o, sp \Rightarrow l_o$ **and** $c_o, sp \Rightarrow ll_o$. Given for a rule of type $c_o, sp \Rightarrow l_o$ ($c_o, sp \Rightarrow ll_o$) are a class $c \in C$, a term $s \in \mathcal{U} \cup \mathcal{B}$, a predicate $p \in \mathcal{P}$, and a (localized) linguistic pattern l.

$$c \in c_e \wedge (s, p, e) \in G\} \Rightarrow l \in l_e \qquad (c_o, sp \Rightarrow l_o)$$
$$c \in c_e \wedge (s, p, e) \in G\} \Rightarrow l \in l_e^{c,p,d} \qquad (c_o, sp \Rightarrow ll_o)$$

Meaning: Given an entity e that is an instance of the class c and given that the term s is in relation p with e, predict that the text that describes e contains the (localized) linguistic pattern l.

Example for rule pattern $c_o, sp \Rightarrow l_o$:

$dbo{:}WineRegion \in c_e \wedge (dbr{:}Mendocino\_County\_wine,$	sup(A)= 11	AllConf(A,B)≈ 0.92
$dbp{:}subRegions, e) \in G$	sup(B)= 12	Coherence(A,B)≈ 0.48
$\Rightarrow$ "Mendocino County California $\in l_e$	sup(AB)= 11	Cosine(A,B)≈ 0.96
	P(B|A)≈ 0.92	IR(A,B)≈ 0.08
	P(A|B)≈ 1	Kulczynski(A,B)≈ 0.96
		MaxConf(A,B)≈ 1

Meaning: Given an entity e that is an instance of the class *dbo*:$WineRegion$ and where the entity *dbr*:$Mendocino\_County\_wine$ is in the relation *dbp*:*subRegions* with e, the rule predicts that the linguistic pattern "Mendocino County California" occurs in the text about the entity e.

Example for rule pattern $c_o, sp \Rightarrow ll_o$:

$dbo{:}Airline \in c_e \wedge (dbr{:}Lufthansa, dbo{:}subsidiary, e) \in G$	sup(A)= 11	AllConf(A,B)≈ 0.09
$\Rightarrow$ "subsidiary of" $\in l_e^{dbo:Airline, dbo:subsidiary, so}$	sup(B)= 64	Coherence(A,B)≈ 0.08
	sup(AB)= 6	Cosine(A,B)≈ 0.23
	P(B|A)≈ 0.09	IR(A,B)≈ 0.77
	P(A|B)≈ 0.55	Kulczynski(A,B)≈ 0.32
		MaxConf(A,B)≈ 0.55

Meaning: Given an entity e that is an instance of the class *dbo*:*Airline* and where the entity *dbr*:*Lufthansa* is in the relation *dbo*:*subsidiary* with e, the rule predicts that the localized linguistic pattern "subsidiary of" (which is localized to the class *dbo*:*Airlines* and the predicate *dbo*:*subsidiary*) occurs in the text about the entity e.

Rule Pattern $c_o, s \Rightarrow l_o$. Given for a rule of type $c_o, s \Rightarrow l_o$ are a class $c \in C$, a term $s \in \mathcal{U} \cup \mathcal{B}$, and a linguistic pattern l.

$$c \in c_e \wedge \exists p \in \mathcal{U} : (s, p, e) \in G\} \Rightarrow l \in l_e \qquad (c_o, s \Rightarrow l_o)$$

Meaning: Given an entity e that is an instance of the class c and given that the term s is in some with e, predict that the text that describes e contains the (localized) linguistic pattern l.

Example for rule pattern $c_o, s \Rightarrow l_o$:

$dbo{:}Horse \in c_e \wedge \exists p : (dbr{:}Orme\_(horse), p, e) \in G$	sup(A)= 14	AllConf(A,B)≈ 0.43
	sup(B)= 11	Coherence(A,B)≈ 0.24
$\Rightarrow$ "English Thoroughbred racehorse" $\in l_e$	sup(AB)= 6	Cosine(A,B)≈ 0.48
	P(B|A)≈ 0.55	IR(A,B)≈ 0.16
	P(A|B)≈ 0.43	Kulczynski(A,B)≈ 0.49
		MaxConf(A,B)≈ 0.55

Meaning: Given an entity e that is an instance of the class *dbo*:$Horse$ and where the entity *dbr*:$Orme\_(horse)$ is in some relation with e, the rule predicts that the linguistic pattern "English Thoroughbred racehorse" occurs in the text about the entity e.

Rule Patterns $c_o, p \Rightarrow l_o$ **and** $c_o, p \Rightarrow ll_o$. Given for a rule of type $c_o, p \Rightarrow l_o$ $(c_o, p \Rightarrow ll_o)$ are a class $c \in C$, a predicate $p \in \mathcal{P}$, and a (localized) linguistic pattern l.

$$c \in c_e \wedge \exists s \in \mathcal{U} \cup \mathcal{B} : (s, p, e) \in G \Rightarrow l \in l_e \qquad (c_o, p \Rightarrow l_o)$$
$$c \in c_e \wedge \exists s \in \mathcal{U} \cup \mathcal{B} : (s, p, e) \in G \Rightarrow l \in l_e^{c,p,d} \qquad (c_o, p \Rightarrow ll_o)$$

Meaning: Given an entity e that is an instance of the class c and given that some term is in relation p with e, predict that the text that describes e contains the (localized) linguistic pattern l.

Example for rule pattern $c_o, p \Rightarrow l_o$:

$dbo{:}Wrestler \in c_e \wedge \exists s : (s, dbp{:}bronze, e) \in G$	sup(A)= 30	AllConf(A,B)≈ 0.47
	sup(B)= 29	Coherence(A,B)≈ 0.24
$\Rightarrow$ "bronze medal" $\in l_e$	sup(AB)= 14	Cosine(A,B)≈ 0.47
	P(B|A)≈ 0.48	IR(A,B)≈ 0.02
	P(A|B)≈ 0.47	Kulczynski(A,B)≈ 0.47
		MaxConf(A,B)≈ 0.48

Meaning: Given an entity e that is an instance of the class *dbo*:$Wrestler$ and where some entity is in the relation *dbp*:*bronze* with e, the rule predicts that the linguistic pattern "bronze medal" occurs in the text about the entity e.

Example for rule pattern $c_o, p \Rightarrow ll_o$:

$dbo{:}Crustacean \in c_e \wedge \exists s : (s, dbp{:}superfamilia, e) \in G$	sup(A)= 33	AllConf(A,B)≈ 0.42
	sup(B)= 15	Coherence(A,B)≈ 0.29
$\Rightarrow$ "is a superfamily" $\in l_e^{dbo:Crustacean,dbp:superfamilia,so}$	sup(AB)= 14	Cosine(A,B)≈ 0.63
	P(B|A)≈ 0.93	IR(A,B)≈ 0.53
	P(A|B)≈ 0.42	Kulczynski(A,B)≈ 0.68
		MaxConf(A,B)≈ 0.93

Meaning: Given an entity e that is an instance of the class *dbo*:$Crustacean$ and where some entity is in the relation *dbp*:$superfamilia$ with e, the rule predicts that the localized linguistic pattern "is a superfamily" (which is localized to the class *dbo*:$Crustacean$ and the predicate *dbo*:$superfamilia$) occurs in the text about the entity e.

HISTORIAE, History of Socio-Cultural Transformation as Linguistic Data Science. A Humanities Use Case

Florentina Armaselu[1]
Centre for Contemporary and Digital History (C^{2}DH), University of Luxembourg, Luxembourg

Elena-Simona Apostol
Department of Computer Science and Engineering, Faculty of Automatic Control and Computer, University Politehnica of Bucharest, Romania

Anas Fahad Khan
Institute for Computational Linguistics «A. Zampolli»,
National Research Council of Italy, Pisa, Italy

Chaya Liebeskind
Department of Computer Science, Jerusalem College of Technology, Israel

Barbara McGillivray
Theoretical and Applied Linguistics, Faculty of Modern and Medieval Languages and Linguistics, University of Cambridge, UK
The Alan Turing Institute, London, UK

Ciprian-Octavian Truică
Department of Computer Science and Engineering, Faculty of Automatic Control and Computer, University Politehnica of Bucharest, Romania

Giedrė Valūnaitė Oleškevičienė
Institute of Humanities, Mykolas Romeris University, Vilnius, Lietuva

Abstract

The paper proposes an interdisciplinary approach including methods from disciplines such as history of concepts, linguistics, natural language processing (NLP) and Semantic Web, to create a comparative framework for detecting semantic change in multilingual historical corpora and generating diachronic ontologies as linguistic linked open data (LLOD). Initiated as a use case (UC4.2.1) within the COST Action *Nexus Linguarum, European network for Web-centred linguistic data science*, the study will explore emerging trends in knowledge extraction, analysis and representation from linguistic data science, and apply the devised methodology to datasets in the humanities to trace the evolution of concepts from the domain of socio-cultural transformation. The paper will describe the main elements of the methodological framework and preliminary planning of the intended workflow.

2012 ACM Subject Classification Computing methodologies → Semantic networks; Computing methodologies → Ontology engineering; Computing methodologies → Temporal reasoning; Computing methodologies → Lexical semantics; Computing methodologies → Language resources; Computing methodologies → Information extraction

Keywords and phrases linguistic linked open data, natural language processing, semantic change, diachronic ontologies, digital humanities

Digital Object Identifier 10.4230/OASIcs.LDK.2021.34

Author Contributions F.A., Sections 1, 2.1, 2.2, 2.5, 2.6, 2.7, 3; E.S.A., Section 2.4; A.F.K., Section 2.3; C.L., Section 1.3; B.M., Sections 1.3, 2.4; C.O.T., Section 2.4; G.V.O., Sections 1.3, 1.4. All the authors critically revised and approved the final version submitted to the LDK 2021 proceedings.

[1] florentina.armaselu@uni.lu

3rd Conference on Language, Data and Knowledge (LDK 2021).
Editors: Dagmar Gromann, Gilles Sérasset, Thierry Declerck, John P. McCrae, Jorge Gracia, Julia Bosque-Gil, Fernando Bobillo, and Barbara Heinisch; Article No. 34; pp. 34:1–34:13

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Use case description

1.1 Contextualisation

Semantic change has been studied so far within various disciplines and research fields, including the history of concepts and philosophy, linguistics, natural language processing (NLP) and the Semantic Web. Despite growing interest in the topic, which requires multiple perspectives and an interdisciplinary approach, there is no unified view and not enough dialogue on the subject, and different disciplines seem to make use of different interpretations and theoretical notions when dealing with it. Our proposal, called *HIstory of Socio-culTural transfORmation as linguistIc dAta sciEnce* (HISTORIAE) with reference to Tacitus's *Historiae* and nowadays interconnected cloud of linked data, aims at bridging the gap and combining approaches from these fields to create a comparative methodological framework for detecting semantic change in multilingual text collections and for generating corpus-based diachronic ontologies as linguistic linked open data (LLOD). The area of application of this proposal spans the digital humanities (DH), with a focus on the history of socio-cultural transformation in Europe and other regions, and emerging trends in knowledge extraction, analysis and representation from linguistic data science. These directions are noteworthy for current research, given the increasing use of digital and Web technologies in almost all the sectors of human activity and the need for a better understanding of their impact on cultural assets, within a broader historical, technological and data-aware context. It is expected that the project outcomes may also be applied to other domains.

HISTORIAE will address the following research questions. (1) Which insights does the study of semantic change help generate in the history of socio-cultural transformation? (2) Can the applied methodology inform us about the interrelation between linguistic, social and cultural innovation over time, and the socio-cultural roots of innovation? (3) What may be learned about the combination of human and machine agency in the process of construction and dissemination of knowledge, and of explaining the underlying mechanisms?

Throughout this paper, the term "semantic change" will generally refer to a change in meaning, either of a lexical unit (word or expression) or of a concept (a complex knowledge structure that can encompass one or more lexical units, as well as relations among them and with other concepts). The contribution of the proposal to the fields of digital humanities and linguistic data science will therefore consist of a workflow prototype based on a combined approach to semantic change, implying data-related and theoretical enquiry, corpus-based analysis and ontology building, and reflection and documentation on the process as a whole. Since the project is still in an early stage, the paper will limit its scope to the following points: (1) the main elements of the HISTORIAE proposal (goals, tasks, datasets, concepts, challenges); (2) exploratory, preliminary planning and research directions of the intended workflow (theoretical models, formalisms and modalities for detecting and representing semantic change, ontology generation, publication, interpretation and documentation).

1.2 Goals, tasks, methods

HISTORIAE builds on the humanities use case (UC4.2.1) initiated as part of the working group "Use cases and applications" within *Nexus Linguarum, European network for Web-centred linguistic data science*, a COST Action (CA18209) [2] running from 2019 to 2023. While UC4.2.1 will be carried out within *Nexus Linguarum* as a pilot, it is intended to further

[2] `https://nexuslinguarum.eu/`

develop the idea within HISTORIAE as a larger interdisciplinary research project, if funding resources are obtained. The main goal of UC4.2.1 is to create a comparative methodological framework for tracing the "histories" or evolution of concepts in different languages and humanities fields (history, literature, philosophy, religion, etc.) and generate a sample of multilingual LLOD ontologies to represent semantic change by using NLP and Semantic Web technologies. Starting from the hypothesis that historical realities are always reflected in language and its manifestations, irrespective of the specific language, it is assumed that such a methodology will allow for comparative transnational and linguistic standpoints and for new insights into the interconnections between language and historical and cultural context over time and space through linguistic data science.

Six tasks (T1-T6) have been designed for the use case (at the time of writing, T1 is completed, T2, 3, 6 ongoing, T4, 5 not yet started). T1 deals with the identification of potential datasets, concepts and languages to be used in the study. T2 has as objectives to draw on the state-of-the-art in LLOD and NLP methods, tools and data, with a focus on the humanities, and provide a terminological and methodological ground for the construction of a theoretical model to detect and represent semantic change in multilingual historical text collections. T3 consists of the selection of the datasets, periods and time span granularity (years, decades, centuries) as well as data preparation (e.g. conversion from one format to another, grouping by time period). T4 and T5 are dedicated to testing and implementing various methods for semantic change detection, representation and publication as LLOD ontologies based on the selected datasets. Finally, T6 is intended to result interpretation and documentation of the process by making use of explainable AI (XAI) techniques, and to a set of guidelines describing the methodology derived from the use case. More details about the methods considered for further investigation are presented in Section 2.

1.3 Datasets, languages, time span

At the initial stage of the study (T1), we identified several datasets, described below, covering a substantial time span and variety of languages such as Latin, Ancient Greek, Hebrew, French, German, Luxembourgish and Old Lithuanian. The LatinISE corpus [32] contains over 10 million word tokens and covers a wide range of genres (e.g. comedy, tragedy, poetry, essays, letters, narrative, oratory, philosophy, religion, law) spanning from the 2nd century BC to the beginning of the 21st century CE. The corpus is lemmatised, part-of-speech (POS)-tagged and searchable through the Sketch Engine corpus query tool. The Ancient Greek corpus Diorisis [50] covers the Ancient Greek literary tradition, from the 8th century BCE to the 5th century CE, and consists of 820 texts (10,206,421 word tokens), which are lemmatised and POS-tagged. Various genres are represented, such as literature (poetry, drama), philosophy, narrative (historiography, biography, mythography), religion (hymns, Jewish and Christian scriptures, homilies), technical literature (medicine, mathematics, natural science, geography, astronomy, politics, rhetoric, art history, literary criticism, grammar), and letters. The Hebrew dataset Responsa [28] includes rabbinic comments on daily issues (law, health, commerce, marriage, education, Jewish customs) and covers the time range from the 11th century until now. It contains 76,710 articles and about 100 million word tokens and can be browsed and searched via a dedicated Web interface. The National Library of Luxembourg (BnL) Open Data collection [12] comprises historical newspapers and monographs (literature, history, philosophy, geography, pedagogy, religious matters, etc.) from the public domain, in French, German and Luxembourgish. It spans two overlapping periods, 1841-1878 (newspapers) and 1690-1918 (monographs). The dataset counts 23,663 processed newspaper issues (510,505 extracted articles), segmented at the level

of individual articles, sub-articles and paragraphs, and 504 processed monographs (33,477 extracted chapters). The Lithuanian dataset Sliekkas [16], which is still under construction, includes Old Lithuanian texts (religious – prayers, catechisms, hymnals, and sermons, as well as prose and poetry), dated between 16th and 18th centuries, with annotations (structural, paleographic, textological, lexical, and grammatical) and facsimile reproductions of the original (ca. 10 million text words).

1.4 Concepts

Tracing the history of concepts is not a new field of research. Various studies have been dedicated to this area, implying different approaches and domains of application such as political, encyclopaedic, legal and biomedical [51], history of philosophy and of science [4], historical research [13] and digital preservation [47]. However, studies in cultural and conceptual history (Begriffsgeschichte) [1], [41] have pointed out the challenges in examining language in its interaction with social, political and cultural transformations from the real world, and the need for a comparative, transnational and interdisciplinary approach to understand the complexities of this type of relationship.

Our proposal aims at creating comparative standpoints to trace the history of concepts in the domain of socio-cultural transformation at a transnational level. The particularity of the contribution mainly consists in combining various approaches and resources from areas such as the history of concepts and linguistic data science, considered together with this domain of application needing further exploration and insight within a digital framework. A series of semantic fields have been identified for this purpose. Examples of such fields include: geo-political and cultural entities (Europe, West, East, etc.), education, sciences, technology and innovations, social and societal processes (migration, urbanisation, modernisation, globalisation), state and citizenship, beliefs, values and attitudes (e.g. religion, democracy, political participation), economy, health and well-being, everyday life, family and social relations, time and collective memory, work and leisure, customs and traditions, literature and philosophy. Moreover, the study will focus on serendipity and the discovery of "turning points", concepts that underwent significant semantic changes at certain points in time, as indicators of shifting or emerging trends in the area of socio-cultural transformation.

The identified datasets will allow for further research in the history of concepts and within the considered domain. It should be noted that the feature of aligning Old Lithuanian translations with their sources in Latin, German or Polish, or comparing the selected languages, will enable us to identify and assess possible mutual cultural influences and look for emerging shared literary, religious and cultural concepts.

1.5 Challenges

The proposal encompasses a number of challenges. (1) Dataset-related ones mainly referring to aspects such as differences in format, time span, genres, size and availability. (2) Workflow-related ones residing in heterogeneous approaches and workflow components to be integrated into a coherent pipeline; under-developed or not yet existing resources (tools, methods, models, formalisms) to deal with certain languages or aspects of semantic change and diachronic ontology generation and publishing. (3) Domain-related ones generally pointing to questions such as adequacy of the considered datasets and linguistic data science methods and tools for tracing a history of concepts that reflects socio-cultural transformation and provides comparative historical, linguistic and cultural insights from a transnational perspective. Possible modalities of addressing these challenges are described in the following section.

2 Workflow planning

For the implementation of the use case, we propose a workflow composed of seven task categories as illustrated below (Figure 1).

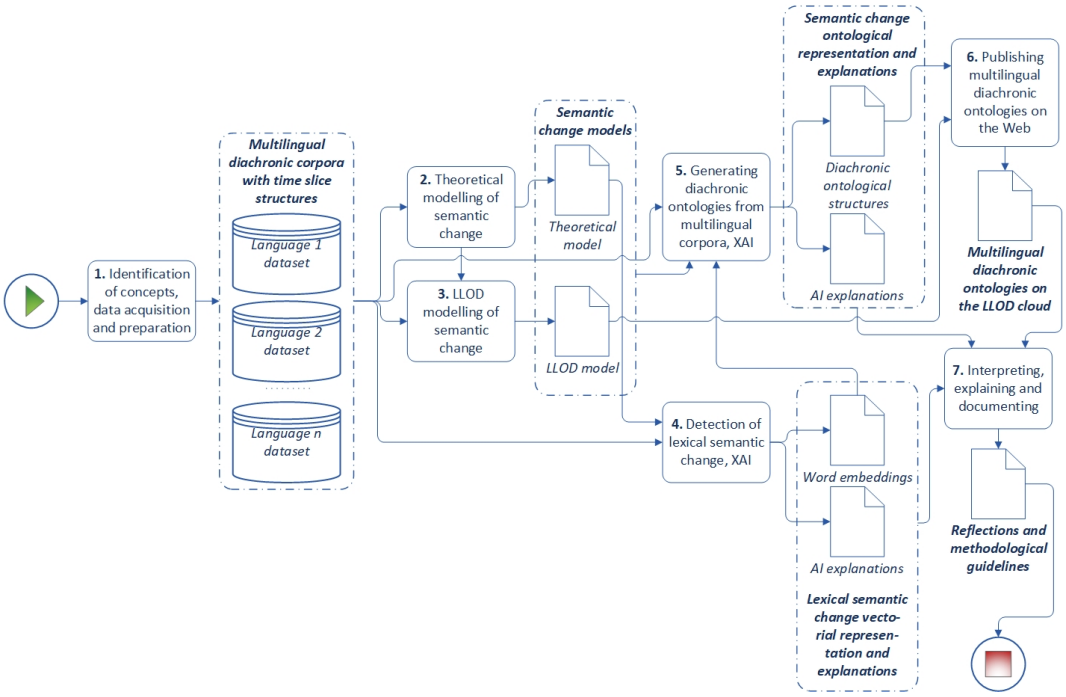


Figure 1 HISTORIAE workflow. Rounded rectangles: task categories; folded-corner rectangles: data types created in the process; dotted rectangles: groups of conceptually associated elements.

2.1 Identification of concepts, dataset acquisition and preparation

An in-depth analysis for the identification of relevant concepts, semantic fields and datasets (T1) (see 1.3, 1.4) to be used in the study will be performed. The core dataset identified so far will be assessed and possibly expanded. The selection criteria mainly pertain to the availability of data, and temporal, geographical, linguistic and thematic coverage enabling a historical, comparative perspective on the topic of socio-cultural transformation. We expect additional datasets, genres and languages to be included in the expanded version of the study (i.e. in Bulgarian, English, Polish, Romanian and Slovene). In order to further extend the time coverage to more recent periods, multilingual contemporary data in open access will be considered, such as Wikimedia downloads, the Digital Corpus of the European Parliament (DCEP) and a collection of trained Twitter word embeddings in English. Preliminary data preparations will be necessary for the whole collection (T3), such as normalisation of old forms, extraction of textual content by genre or language from XML, and segmentation of the corpora by time slice (e.g. year, decade, century) for diachronic analysis.

2.2 Theoretical modelling of semantic change

From a theoretical point of view, four research directions have been identified and will be further explored (T2) as starting points in designing the theoretical model to approach semantic change. It is assumed that such a theoretical model may be combined in the workflow with elements of LLOD formalisation and NLP-based detection of lexical semantic change and diachronic ontology generation (Sections 2.3, 2.4 and 2.5).

Within the theory of lexical semantics, [15] identifies two main classifications of semantic change that include semasiological mechanisms (meaning-related), with semasiological innovations endowing existing words with new meanings, and onomasiological (or "lexicogenetic") (naming-related) mechanisms, with onomasiological innovations expressing meaning through new or alternative lexical items. [15] also draws attention to semantic approaches, in the lineage of distributional semantics, inspired by Firth and Harris, that display a certain affinity with current usage-based approaches and distributional corpus analysis. From the field of intellectual history, theory of knowledge organisation and Semantic Web, two formal descriptions of conceptual change have been retained for further analysis. One is proposed by [27] and asserts that a concept is composed by two parts, the "core" and the "margin", based on context-nonspecific and context-specific features. This model allows for a variety of possibilities, from conceptual continuity, implying core stability and different degrees of margin variability, to conceptual replacement, when the core itself is affected by change. The other formalisation, developed in [51], defines the meaning of a concept in terms of "intension" (a set of properties), "extension" (a subset of the universe) and "label" (a string of characters). [51] use distance measures, such as Jaccard and Levenshtein, computed for the three aspects to identify conceptual changes.

2.3 Expressing semantic change through LLOD formalisms

One of the aims of the work described in this submission is to model and then publish data about semantic change in the form of one or more diachronic lexico-ontologies in order to integrate together different kinds of relevant information and to make this information available in an accessible and easily re-usable form. The linked open data (LOD) publishing paradigm is ideal for doing this. It offers us a standardised way of making structured data available using the HTTP protocol, as well as giving us the possibility of exposing this data via special endpoints that use the powerful SPARQL query language. The use of a common data framework, the Resource Description Framework (RDF), combined with a number of upper level ontologies and more generalised linked data vocabularies helps to ensure the interoperabilty of data published in this way. As we intend to model (and publish) data about linguistic phenomena as linked data (although this may include information from and relevant to other disciplines such as history) we use the term linguistic linked open data in the current work. In the rest of this section we will give a brief overview (T2) of some of the most relevant vocabularies and datasets for publishing data on semantic change as linguistic linked open data.

The idea would be to create a linked data resource with a lexical component that includes a list of lexical entries and their senses (along with other linguistic information pertaining to for instance the grammatical features of a word) and an ontological or more broadly speaking semantic component that describes the meanings of these senses and, more importantly, the way in which they change over time. The well known OntoLex-Lemon model [31] published by the W3C Ontology-Lexicon group[3] allows for this approach in the case of static senses. However it does not make explicit provision for representing semantic change, nor does it do so for dynamic or time dependent information. This is an issue because the representation of n-ary relations for $n > 2$ can have its drawbacks [52] (in this case relationships which would be most naturally represented as relations with an additional temporal parameter).

[3] `https://www.w3.org/2016/05/ontolex/`

The modelling of dynamic or diachronic lexical information in linked data is still an active area of research and discussion, and it is unlikely that there will be any one-size-fits-all solution. One approach has been proposed in [23] where word senses are represented as perdurants, that is, entities with an extension in time which can have temporal parts. This strategy is also being adopted in the soon to be published ISO Standard ISO 24613-3 which consists of a diachronic module for the Lexical Markup Framework (LMF) [43]. In the case of LLOD the perdurant solution has the advantage, among other things, of allowing the use of certain built-in Web Ontology Language constructs which facilitate automated reasoning on RDF datasets.

2.4 Detecting lexical semantic change

There is a growing body of research on computational methods for detecting lexical semantic change automatically, recently surveyed in [48] and [26].

Word representations that employ semantics, syntax, and context to create vectors are used in current literature to successfully compute semantic change using distance metrics (e.g., cosine, Levenshtein) [46]. These vectors are built using shallow neural networks, and, although they use different architectures to create lexical representation for textual data, are known collectively as word embeddings [34, 38, 37, 6, 35]. Although similar concepts have similar representations, word embeddings cannot detect correctly the semantic changes that appear over time if they are not trained specifically for this task. Thus, in current literature new methods for building word embeddings to detect semantic changes have been proposed. In [18], the authors correlate word embeddings with temporal-spatial information to create condition-specific embeddings. Another method uses hyperbolic embeddings to map partial graphs into low-dimensional, continuous hierarchical spaces to build diachronic semantic hyperspaces for four scientific topics [5]. The current approaches are prone to anomalies and direct human intervention is required to make correct assessments about the results. Thus, new anomaly detection methods that employ unsupervised machine and deep learning are required to alleviate the need for expert validation.

Word embeddings are used with machine translation architectures, e.g., long short-term memory networks (LSTM)-based sequence to sequence models [49], to measure the semantic change of words by tracking their evolution over time in a sequential manner. These approaches seem promising and new deep learning architectures for machine translation can be developed for the task of determining semantic change, e.g., variational autoencoders (VAE) [24] and generative adversarial networks (GAN) [29].

The SemEval 2020 shared task on Unsupervised Lexical Semantic Change Detection [46] has provided the current state-of-the-art in the field, with evaluation results relative to 21 systems evaluated on two subtasks in four languages (English, German, Latin and Swedish). In this task, systems based on word type embeddings outperformed token embeddings on both subtasks, but the potential of token embeddings is yet to be fully explored. [46] also found a strong effect of frequency in the systems based on type embeddings, and a strong correlation between change scores and polysemy. Both these factors should be further explored and taken into account in future studies and implementations.

Recently transformers-based models have also been considered for lexical semantic change detection. Most solutions that fall into this category use BERT (Bidirectional Encoder Representations from Transformers) [10]. BERT employs a bidirectional attention mechanism to learn the contextual relations. Pre-trained BERT models have been used in both unsupervised [17, 22] and supervised [42] semantic shift tracing solutions. Another transformer that was applied to semantic change detection is ELMo (Embeddings from Language

Models) [39]. ELMo provides faster training and inference compared with BERT. Because of this, it is much easier to train the models with ELMo on specific datasets, and not use pre-trained models. A comparison between ELMo and BERT in semantic change detection for the Russian language is presented in [42]. By analysing the results presented in the state-of-the-art solutions, we can conclude that transformers enhance the semantic change detection task. For this purpose, we are considering experimenting with other, more recent transformers (T4). One candidate is DistilBERT [45] which is used to pre-train a smaller general-purpose language representation model by reducing the size of BERT. RoBERTa [30] is another candidate. This model improves BERT's language masking strategy by adjusting several hyperparameters.

2.5 Generating diachronic ontologies from corpora

Another area of interest for the study is that of ontology learning from text, surveyed in [21, 2]. An influential model used in many applications, the so-called "ontology learning layer cake" [7], proposes six steps or layers for ontology acquisition, dedicated to *terms*, *synonyms*, *concepts*, *concept hierarchies*, *relations* and *rules*. [7] list different techniques from various fields of research to achieve this task. The first three subtasks include information retrieval methods for term extraction, synonym acquisition from lexical-semantic resources (e.g. WordNet), text corpora and the Web based on synset relations, Harris' distributional hypothesis and statistical information measures, and concept induction through definition learning (*intension*), deriving instances from named entity hierarchies (*extension*) and linguistic realisation (*terms*) (see also Section 2.2). For the last three subtasks, [7] mention taxonomic (*is-a*) and non-taxonomic relation extraction based on hierarchical clustering algorithms as well as statistical and linguistic analysis of syntactic structure and dependencies, and ontological rules learning from text using lexical entailment. This framework will serve as a starting point for designing this workflow phase (T5), possibly in combination with deep learning approaches (e.g. word2vec, LSTMs), human-based evaluation and post-processing that seem promising for ontology learning goals according to [53], [21] and [2].

For more specific objectives in generating diachronic ontologies from historical corpora included in the use case, additional methods will be assessed, such as distributional semantic models and "hubs and authorities" [20], hyperbolic embeddings [5], "peak detection" in time series and word and event "projected embeddings" [44] or vector representation of concept signatures [19]. Possible integration with recent advances in transformers-based models and other state-of-the-art NLP methods for lexical semantic change detection (Section 2.4) will be considered as well.

2.6 Publishing diachronic ontologies as LL(O)D

Unlike synchronic ontologies that ignore the historical perspective, diachronic ontologies allow us to capture the temporal dimension of concepts and investigate gradual semantic changes and concept evolution through time [20]. Since the goal of the study is to produce a sample of diachronic ontologies represented and published on the Web as LL(O)D (T5), a set of existing methods and tools for acquiring (Section 2.5) and converting ontological structures into Semantic Web formalisations will be evaluated together with modalities of expressing semantic change through LLOD formalisms (Section 2.3). One of the systems often cited as a reference is Text2Onto [9], an ontology learning framework that converts learned knowledge into a Probabilistic Ontology Model (POM) translatable into various ontology representation languages such as RDFS, OWL and F-Logic. Other tools, e.g. LODifier [3] and OntoGain [11],

can extract entities and relations from text and produce RDF representations linked to the LOD cloud using DBpedia and WordNet 3.0 vocabularies, or transform the acquired ontology into standard OWL statements. More specialised tools, such as converters, allow for making linked data in RDF format out of CSV files (CoW [33]), converting language resources into LLOD (LLODifier [8]) or developing complex transformation pipelines for converting heterogeneous linguistic resources to RDF (Fintan [14]).

2.7 Interpreting, explaining and documenting the process

For the interpretative approach, we will take into account linguistic, cultural and historical aspects of linguistic innovation and its temporal and referential complexity starting from the theoretical model of the *concept – reality relationship*, based on the four combinations of synchronous and asynchronous concept and reality change vs. stability over a period of time [25]. The implementation of the proposed workflow will include qualitative analysis and XAI components (in phases 2.4 and 2.5) for interpreting the results, explaining, documenting and reflecting on the process (T6). As starting points we will consider the four principles of explainable AI systems [40] and insight from the social sciences in designing this type of components [36]. The outcome will consist of comparative insights into the history of socio-cultural transformation, and in particular the interconnection between linguistic innovation and social and cultural innovation, and their evolution over time. It will also contain methodological guidelines and reflections on the hybridisation of human and algorithmic approaches and the role of AI from the sociology of knowledge perspective, in order to understand how these technologies are changing our modes of producing, disseminating and consuming knowledge.

3 Conclusion and future work

The paper presents a use case and further development proposal for detecting and representing semantic change by means of NLP and LL(O)D technologies applied to multilingual historical datasets and various humanities areas in order to trace the evolution of concepts in the domain of socio-cultural transformation. A set of challenges has been identified, mainly related to the heterogeneity of the datasets and approaches, as well as the complexity of the application domain and of constructing comparative standpoints to derive historical, linguistic and cultural insight from a transnational perspective. Given the early stage in the use case development, the proposal does not present experimental descriptions and results but a set of methodologies and tools to be further examined, tested and evaluated within the planned workflow. It is expected that some of the defined challenges will be addressed by combining various approaches in linguistic data science, e.g. for theoretical modelling, detection and representation of semantic change and diachronic ontology learning, as well as documentation and reflection on the process itself making use of human- and AI-based explainability. The dataset diversity may provide opportunities for reflection on the gaps in the data and the possibilities for alleviating incompleteness and uncertainty by a modular, expansible design and an explainability- and discovery-based architecture. The next steps of the study will therefore consist in testing the hypotheses formulated in the present proposal to confirm or disconfirm their validity and create the bases for the construction of the comparative framework and workflow prototype for detecting and representing semantic change through NLP and LLOD technologies.

References

1 Alessandro Arcangeli. *Cultural History. A Concise Introduction.* Routledge, 1 edition, 2012. doi:10.4324/9780203789247.

2 Muhammad Nabeel Asim, Muhammad Wasim, Muhammad Usman Ghani Khan, Waqar Mahmood, and Hafiza Mahnoor Abbasi. A Survey of Ontology Learning Techniques and Applications. *Database*, 2018, January 2018. doi:10.1093/database/bay101.

3 Isabelle Augenstein, Sebastian Padó, and Sebastian Rudolph. LODifier: Generating Linked Data from Unstructured Text. In Elena Simperl, Philipp Cimiano, Axel Polleres, Oscar Corcho, and Valentina Presutti, editors, *The Semantic Web: Research and Applications*, volume 7295 of *Lecture Notes in Computer Science*, page 210–224. Springer Berlin Heidelberg, 2012. doi:10.1007/978-3-642-30284-8_21.

4 Arianna Betti and Hein van den Berg. Modelling the History of Ideas. *British Journal for the History of Philosophy*, 22(4):812–835, 2014. doi:10.1080/09608788.2014.949217.

5 Yuri Bizzoni, Marius Mosbach, Dietrich Klakow, and Stefania Degaetano-Ortlieb. Some Steps Towards the Generation of Diachronic WordNets. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 55–64, 2019. URL: https://www.aclweb.org/anthology/W19-6106.

6 Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017. doi:10.1162/tacl_a_00051.

7 P. Buitelaar, P. Cimiano, and B. Magnini. Ontology Learning from Text: An Overview. In *Ontology Learning from Text: Methods, Evaluation and Applications*, volume 123, pages 3–12. IOS Press, 2005.

8 Philipp Cimiano, Christian Chiarcos, John P. McCrae, and Jorge Gracia. Linguistic Linked Data in Digital Humanities. In *Linguistic Linked Data. Representation, Generation and Applications*, pages 229–262. Springer International Publishing, 1 edition, 2020. URL: https://www.springer.com/gp/book/9783030302245.

9 Philipp Cimiano and Johanna Volker. Text2Onto. A Framework for Ontology Learning and Data-driven Change Discovery. *Natural Language Processing and Information Systems: 10th International Conference on Applications of Natural Language to Information Systems, NLDB 2005, Alicante, Spain, June 15 - 17, 2005; proceedings. Lecture Notes in Computer Science, 3513. Montoyo A, Munoz R, Metais E (Eds); Springer: 227-238*, 2005.

10 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics*, pages 4171–4186. Association for Computational Linguistics, 2019. doi:10.18653/v1/N19-1423.

11 Euthymios Drymonas, Kalliopi Zervanou, and Euripides G. M. Petrakis. Unsupervised Ontology Acquisition from Plain Texts: The OntoGain System. In Christina J. Hopfe, Yacine Rezgui, Elisabeth Métais, Alun Preece, and Haijiang Li, editors, *Natural Language Processing and Information Systems*, volume 6177 of *Lecture Notes in Computer Science*, page 277–287. Springer Berlin Heidelberg, 2010. doi:10.1007/978-3-642-13881-2_29.

12 Maud Ehrmann, Matteo Romanello, Simon Clematide, Phillip Ströbel, and Raphaël Barman. Language Resources for Historical Newspapers: the Impresso Collection. In *Proceedings of the 12th Language Resources and Evaluation Conference*. European Language Resources Association (ELRA), 2020.

13 Antske Fokkens, Serge Ter Braake, Isa Maks, and Davide Ceolin. On the Semantics of Concept Drift: Towards Formal Definitions of Semantic Change. *Drift-a-LOD@EKAW*, 2016.

14 Christian Fäth, Christian Chiarcos, Björn Ebbrecht, and Maxim Ionov. Fintan - Flexible, Integrated Transformation and Annotation eNgineering. In *Proceedings of the 12th Conference on Language Resources and Evaluation*, page 7212–7221. European Language Resources Association (ELRA), licensed under CC-BY-NC, May 2020.

15 Dirk Geeraerts. *Theories of lexical semantics.* Oxford University Press, 2010.

16 Jolanta Gelumbeckaite, Mindaugas Sinkunas, and Vytautas Zinkevicius. Old Lithuanian Reference Corpus (SLIEKKAS) and Automated Grammatical Annotation. *J. Lang. Technol. Comput. Linguistics*, 27(2):83–96, 2012.

17 Mario Giulianelli, Marco Del Tredici, and Raquel Fernández. Analysing Lexical Semantic Change with Contextualised Word Representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3960–3973, Online, 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.365.

18 Hongyu Gong, Suma Bhat, and Pramod Viswanath. Enriching Word Embeddings with Temporal and Spatial Information. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 1–11, Online, November 2020. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/2020.conll-1.1.

19 Jon Atle Gulla, Geir Solskinnsbakk, Per Myrseth, Veronika Haderlein, and Olga Cerrato. Semantic Drift in Ontologies. In *WEBIST 2010, Proceedings of the 6th International Conference on Web Information Systems and Technologies*, volume 2, April 2010.

20 Shaoda He, Xiaojun Zou, Liumingjing Xiao, and Junfeng Hu. Construction of Diachronic Ontologies from People's Daily of Fifty Years. *LREC 2014 Proceedings*, 2014.

21 Vivek Iyer, Mohan Mohan, Y. Raghu Babu Reddy, and Mehar Bhatia. A Survey on Ontology Enrichment from Text. In *The sixteenth International Conference on Natural Language Processing (ICON-2019)*, 2019.

22 Vani Kanjirangat, Sandra Mitrovic, Alessandro Antonucci, and Fabio Rinaldi. SST-BERT at SemEval-2020 Task 1: Semantic Shift Tracing by Clustering in BERT-based Embedding Spaces. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 214–221, 2020.

23 Anas Fahad Khan. Towards the Representation of Etymological Data on the Semantic Web. *Information*, 9(12), November 2018. Publisher: MDPI AG. doi:10.3390/info9120304.

24 Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *International Conference on Learning Representations*, 2014.

25 Reinhart Koselleck. Some Reflections on the Temporal Structure of Conceptual Change. In Willem Melching and Velema Wyger, editors, *Main Trends in Cultural History. Ten Essays*, page 7–16. Rodopi, 1994.

26 Andrey Kutuzov, Lilja Øvrelid, Terrence Szymanski, and Erik Velldal. Diachronic Word Embeddings and Semantic Shifts: A Survey. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1384–1397, Santa Fe, New Mexico, USA, 2018. Association for Computational Linguistics.

27 Jouni-Matti Kuukkanen. Making Sense of Conceptual Change. *History and Theory*, 47(3):351–372, 2008. doi:10.1111/j.1468-2303.2008.00459.x.

28 Chaya Liebeskind and Shmuel Liebeskind. Deep Learning for Period Classification of Historical Hebrew Texts. *Journal of Data Mining and Digital Humanities*, 2020.

29 Jianyi Liu, Yu Tian Ru Zhang, Youqiang Sun, and Chan Wang. A Two-Stage Generative Adversarial Networks With Semantic Content Constraints for Adversarial Example Generation. *IEEE Access*, 8:205766–205777, 2020. doi:10.1109/ACCESS.2020.3037329.

30 Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv:1907.11692 [cs]*, July 2019. arXiv:1907.11692.

31 John P. McCrae, Julia Bosque-Gil, Jorge Gracia, Paul Buitelaar, and Philipp Cimiano. The OntoLex-Lemon Model: Development and Applications. *Electronic Lexicography in the 21st Century. Proc. of eLex 2017 conference, in Leiden, Netherlands*, pages 587–597, September 2017. Publisher: Lexical Computing CZ s.r.o. URL: https://elex.link/elex2017/wp-content/uploads/2017/09/paper36.pdf.

32 Barbara McGillivray and Adam Kilgarriff. Tools for Historical Corpus Research, and a Corpus of Latin. *New Methods in Historical Corpus Linguistics*, 1(3):247–257, 2013.

33 Albert Meroño-Peñuela, Victor de Boer, Marieke van Erp, Willem Melder, Rick Mourits, Ruben Schalk, and Richard Zijdeman. Ontologies in CLARIAH: Towards Interoperability in History, Language and Media. *arXiv*, 2020. arXiv:2004.02845v2.

34 Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In *International Conference on Learning Representations*, pages 1–12, 2013.

35 Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. Advances in Pre-Training Distributed Word Representations. In *International Conference on Language Resources and Evaluation*, pages 52–55, 2018.

36 Tim Miller. Explanation in Artificial Intelligence: Insights from the Social Sciences. *Artificial Intelligence*, 267:1–38, February 2019. doi:10.1016/j.artint.2018.07.007.

37 Maximilian Nickel and Douwe Kiela. Poincaré Embeddings for Learning Hierarchical Representations. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, page 6341–6350, 2017.

38 Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. Association for Computational Linguistics, October 2014. doi:10.3115/v1/D14-1162.

39 Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi:10.18653/v1/N18-1202.

40 P. Jonathon Phillips, Carina A. Hahn, Peter C. Fontana, David A. Broniatowski, and Mark A. Przybocki. *Four Principles of Explainable Artificial Intelligence*. National Institute of Standards and Technology, U.S. Department of Commerce, August 2020. doi:10.6028/NIST.IR.8312-draft.

41 Melvin Richter. *The History of Political and Social Concepts: A Critical Introduction*. Oxford University Press, 1995.

42 Julia Rodina, Yuliya Trofimova, Andrey Kutuzov, and Ekaterina Artemova. ELMo and BERT in Semantic Change Detection for Russian. *CoRR*, abs/2010.03481, 2020. arXiv:2010.03481.

43 Laurent Romary, Mohamed Khemakhem, Fahad Khan, Jack Bowers, Nicoletta Calzolari, Monte George, Mandy Pet, and Piotr Bański. LMF Reloaded. *arXiv preprint*, 2019. arXiv:1906.02136.

44 Guy D. Rosin and Kira Radinsky. Generating Timelines by Modeling Semantic Change. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, page 186–195. Association for Computational Linguistics, 2019. doi:10.18653/v1/K19-1018.

45 Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter. In *Workshop on Energy Efficient Machine Learning and Cognitive Computing*, pages 1–5, 2019.

46 Dominik Schlechtweg, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. SemEval-2020 Task 1: Unsupervised Lexical Semantic Change Detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1–23, 2020.

47 Thanos G Stavropoulos, Stelios Andreadis, Marina Riga, Efstratios Kontopoulos, Panagiotis Mitzias, and Ioannis Kompatsiaris. A Framework for Measuring Semantic Drift in Ontologies. In *CEUR Workshop Proceedings Vol-1695*, September 2016.

48 Nina Tahmasebi, L. Borin, and A. Jatowt. Survey of Computational Approaches to Lexical Semantic Change. *arXiv*, 2018. arXiv:1811.06278.

49 Adam Tsakalidis and Maria Liakata. Sequential Modelling of the Evolution of Word Representations for Semantic Change Detection. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8485–8497. Association for Computational Linguistics, November 2020. doi:10.18653/v1/2020.emnlp-main.682.

50 Alessandro Vatri and Barbara McGillivray. The Diorisis Ancient Greek Corpus: Linguistics and Literature. *Research Data Journal for the Humanities and Social Sciences*, 3(1):55–65, 2018.

51 Shenghui Wang, Stefan Schlobach, and Michel Klein. Concept Drift and How to Identify It. *Journal of Web Semantics First Look*, September 2011. doi:10.2139/ssrn.3199520.

52 Chris Welty, Richard Fikes, and Selene Makarios. A Reusable Ontology for Fluents in OWL. In *FOIS*, volume 150, pages 226–236, 2006.

53 Gerhard Wohlgenannt and Filip Minic. Using word2vec to Build a Simple Ontology Learning System. *International Semantic Web Conference*, page 4, 2016.

An Automatic Partitioning of Gutenberg.org Texts

Davide Picca
University of Lausanne, Switzerland

Cyrille Gay-Crosier
University of Lausanne, Switzerland

Abstract

Over the last 10 years, the automatic partitioning of texts has raised the interest of the community. The automatic identification of parts of texts can provide a faster and easier access to textual analysis. We introduce here an exploratory work for multi-part book identification. In an early attempt, we focus on *Gutenberg.org* which is one of the projects that has received the largest public support in recent years. The purpose of this article is to present a preliminary system that automatically classifies parts of texts into 35 semantic categories. An accuracy of more than 93% on the test set was achieved. We are planning to extend this effort to other repositories in the future.

2012 ACM Subject Classification Computing methodologies; Computing methodologies → Language resources

Keywords and phrases Digital Humanities, Machine Learning, Corpora

Digital Object Identifier 10.4230/OASIcs.LDK.2021.35

1 Introduction

Over the last 10 years, the automatic partitioning of texts has raised the interest of the community [6]. In fact, while humans perform text segmentation smoothly during reading, automatic approaches struggle with the problem of inferring the paragraphemic uses of signs. The need for this type of research is also driven by the compelling use of computational methods for literary texts that often do not meet formatting standards [13, 3, 8, 14]. In fact, such an identification would make a finer textual analysis possible, based on the narrative parts of the text (i.e., direct speech, footnote, etc.). Nonetheless, there is a twofold difficulty in this field: on the one hand, the heterogeneity of the encoding methods, which do not adhere to a general standard, and, on the other hand, the diversity of literary repositories making it more complex to provide a general method that fits any repository. In order to tackle this issue, we introduce here an exploratory work for multi-part book identification. In a first attempt to address the problem, we focus on *Gutenberg.org*[1] which is one of the projects that has received the largest public support in recent years [9, 21, 16]. The purpose of this article is to present a preliminary system that automatically classifies parts of text into 35 semantic categories, listed in Table 1. We are planning to extend this effort to other repositories in the future.

2 Related Work

The tracks proposed by the INEX and ICDAR book structure extraction competitions [6, 15, 7] share with our paper the same general topic. In these tracks, participants are asked to submit automated methods for more accurate identification of text parts such as *Abstract*, *Introduction*, *Methods*, *References*. Nonetheless, with respect to these challenges, our work aims to use a manually pre-defined set of categories, which is more related to the work

[1] `https://www.gutenberg.org/`

proposed by [18]. Our article differs in two main respects: on the one hand, we introduce a finer definition of the structural categories extending them from 10 to 35 and, on the other hand, we focus on classifying the parts of the text rather than classifying the pages themselves. Other authors such as [10, 5, 22] also introduce works whose systems rely on parsing the table of contents rather than relying on the content of the book itself. Most of the contributions analyzing the textual content is related to phrases and paragraphs segmentation [20, 2, 17]. Although the task has a relatively solid tradition, it focuses on identifying a specific part of the book's content without taking into account the 35 categories as shown in Table 1. The choice to consider a broader spectrum of categories has a twofold reason. On the one hand, the frequency of each of these categories (See Figure 1) justifies the interest of counting them as relevant. In this way, we also assume that we cover a sufficiently large number of possibilities should these categories be expanded to include other repositories of literary texts. On the other hand, the choice is motivated by the fact that some minor categories (such as epigraphs or figure captions) play a major role in the study of certain literary and linguistic phenomena. Indeed, a great deal of information relevant to scholars working in the literary field resides in very fine-grained categories. By having introduced some subcategories to the macro-categories defined in the Table 1, even though they are widely less used, we believe we are encouraging scholars in such fields to use this tool for their research.

3 Experiment

3.1 Dataset construction

For the experiment, we rely on the Gutenberg Project repository since it is one of the most used repositories in the Digital Humanities [11, 12, 4], with a variety and well-balanced composition of texts. In fact, it consists of more than 50,000 eBooks (i.e., raw text files) of many different genres, like fiction, poetry, journal articles or scientific papers. A corpus of 169 texts was randomly collected by Project Gutenberg using the DHTK library provided by [19]. The corpus includes texts from different eras, genres and authors, to avoid any bias. Out of 169 texts, 111 have finally been retained, so as to have only texts in English [2]. Each text has been downloaded as a *.txt* file.

An initial manual analysis was performed to identify regular patterns to mark the categories. Then, an automatic file segmentation was applied using regular expressions with the intention of capturing the 35 categories. Finally, an annotator checked the entire dataset for double-checking. On the one hand, the annotator checked the accuracy of the algorithm in capturing each category, and on the other hand, it evaluated the recall of the algorithm in order to check that the algorithm did not miss any relevant categories. Since the task was performed by only one annotator, no measure of agreement between annotators was performed, but each part of the texts was labeled according to one of the categories described in the Table 1 until the entire corpus were labeled. A final distribution of the 35 selected categories is shown in Figure 1.

[2] Dataset and code are freely available here `https://gitlab.com/cgaycro1/gutenberg-files-tagging.git`. A request access can be sent through the gitlab platform

Table 1 Parts of text identified in the corpus, sorted by category.

Gutenberg header/footer	Book header/footer	Section	Editorial	Text	Layout
footer	author	chapter number	caption	date	layout
footer license	bibliography	chapter title	character	direct speech	list
footer start	book info	part number	editorial	paragraph	table
header	book title	part title	footnote	place	
header end	epigraph	section number	note	place and date	
header info	glossary	subtitle	play info	quote	
	index				
	table of contents				

3.1.1 Features Engineering.

A selection of 17 features was used. In order to assess their importance, some of them were manually chosen based on observations during the corpus annotation, others were drawn from the McConnaughey's work [18]. The 17 features can be split into three different groups: *textual features, boolean features and numerical features* as listed here under.

Textual features:
TFIDF: This is the raw text processed using TFIDF method. This is the most common feature used in NLP tasks.

First characters: This feature returns the first five characters of a text, including spaces. This feature seems to be very useful for identifying *titles* and *paragraph*.

Last characters : This feature returns the last five characters of a text, including spaces. As the previous one, this feature seems to be very useful for identifying *titles* and *paragraph*.

Class of next part: This feature returns the target class of the next part of text in the document. Most of the time there exists a repetitive pattern in the classes' sequence.

Class of previous part: This feature returns the target class of the previous part of text in the document.

Boolean features:
Ends with punctuation: This feature returns True if the last character of the text part is a punctuation mark. The parts *paragraph*, *direct speech* and *quote* often end with a punctuation mark.

First word in capital letters: This feature returns True if the first word of the text part is in capital letters. The parts *chapter number*, *part number*, *header end*, *footer end* and *book title* often have their first word in capital letters.

Has asterisk : This feature returns True if there is an asterisk in the text part. The parts *header end*, *footer start* and *layout* often have at least an asterisk.

Has bracket: This feature returns True if there is one bracket in the text part. The parts *footnote*, *note* and *caption* often have at least one bracket.

Has quote : This feature returns True if there is one quotation mark in the text part. The parts *direct speech* and *quote* have at least one quotation mark.

Has reporting verbs: This feature returns True if there is one reporting verb in the text part. Reporting verbs are verbs transmitting the action of speaking, such as "say", "explain" or "think". The part *direct speech* often has one reporting verb.

Numerical features:
Part length: This feature returns the length of the text part as an integer. The parts *paragraph* are often longer than other parts.

Ratio symbol: This feature returns the ratio in the text part between the number of symbols and the total number of characters. Symbols are for example currency symbols and hashtags.

Ratio uppercase : This feature returns the ratio in the text part between the number of uppercase letters and the total number of letters. The parts *header info*, *book title*, *chapter title* and *part title* often have words in uppercase letters.

Ratio word/lemma: This feature returns the ratio between the number of lemmas and the number of words.

Ratio word with first capital letter: This feature returns the ratio in the text part between the number of words with their first letter in uppercase and the total number of words. In English, words in titles begin usually with a capital letter. Therefore, the parts *header info*, *book title*, *chapter title* and *part title* often have words with their first letter in uppercase.

Relative position : This feature returns the relative position of the text part in the document.

3.2 Experiment and Results

We approached the problem as a multi-class classification task. The 102,461 target classes of text found in the 111 texts of the corpus (as described in 3.1) were randomly assigned into a training and a test set, given a ratio of 0.33 with the distribution shown in Figure 1

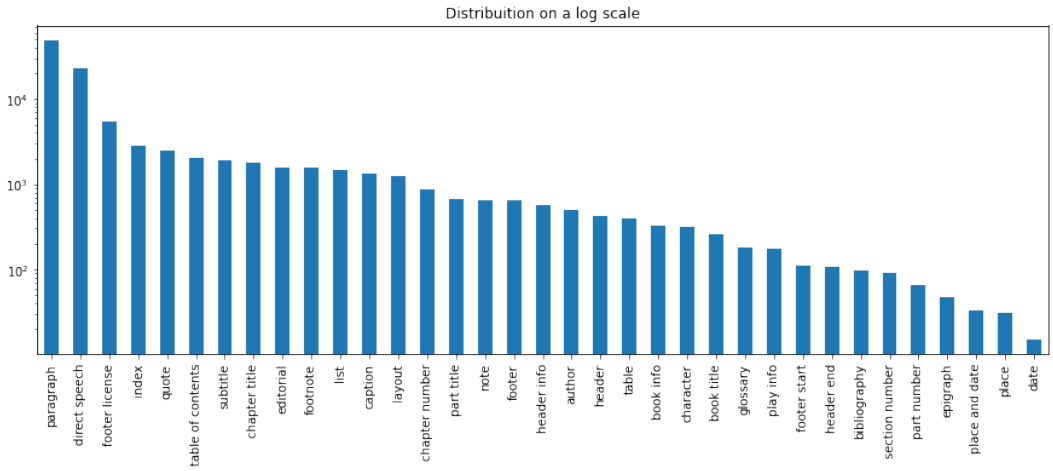


Figure 1 Distribution of target classes on a log scale.

To explore the problem we compared four inherently multiclass classifiers as suggested by [1] and shown in Table 2. Moreover, in order to offset the class imbalance, where possible, we weighed the classes using the following formula: $\frac{|X|}{|T| \times f(T)}$ where X is the cardinality of samples, T is the total number of target classes and f is a function counting the number of elements $t \in T$ whose values lie in successive integer bins.

Three algorithms out of four achieve an overall accuracy of 93% on the test set as shown in Table 2. It can be noticed that, with the exception of the Bernoulli Naive-Bayes classifier, all other classifiers perform encouragingly for each category, crossing an F-Measure of over 90% for almost every class.

The only classes for which the classifiers do not perform well are *dates* and *places*, likely due to the paucity of examples in the training set.

One-feature classifiers and combined-features classifiers were built to compare the performance of individual features on the classification process, similarly as proposed by [20].

Table 2 F-Measure for each classifier based only on one-group feature and all-combined features.

	LinearSVC	KNeighbors	DecisionTree	BernoulliNB
Textual features	**0.940526**	0.876941	0.865288	0.59276
Boolean features	0.584775	0.621152	**0.647887**	0.611008
Numerical features	0.42253	0.721438	**0.765327**	0.483542
All features	**0.953125**	0.946322	0.933369	0.657085

Table 2 shows the F-Measure scores of the three feature groups and all combined features respectively.

Figure 2 shows the F-Measure report for each target class and for each classifier. The classes *date*, *place*, *place and date* are poorly predicted likely due to the lack of support items. While there is room for improvement, the reliability of currently available NERs mitigates the severity of such a negative result. Looking at Table 2, we notice that not all features work equally well. There is a clear distinction between textual and non-textual features. While textual attributes correctly predict almost 9 times out of 10, Boolean features have an overall accuracy of 63% and numeric features hardly get close to 50% for LinerSVC and BernoulliNB.

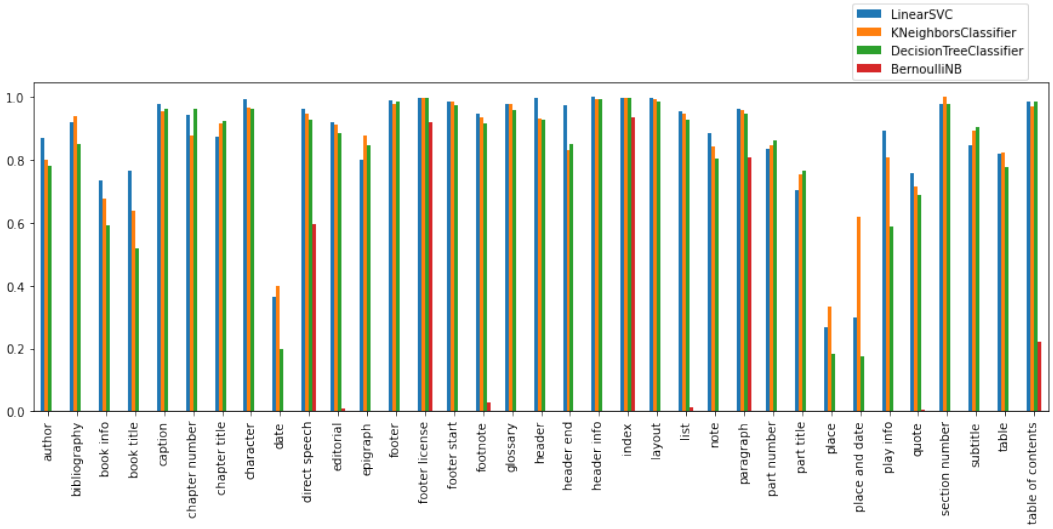


Figure 2 Comparison of F1 Measure on target classes for each classifier.

If we analyze the importance of features by group, we clearly notice that the textual features (see Figure 3) achieve an accuracy of more than 75% and can accommodate almost any class reflecting the importance of the spelling and textual features for this task.

In particular, the textual features *Type of the previous part* and *Type of the next part* help to classify the sections according to their location and the surrounding parts in the text. For example, these two features identify the *index* with almost perfect accuracy, while other textual features do not work well for that specific class.

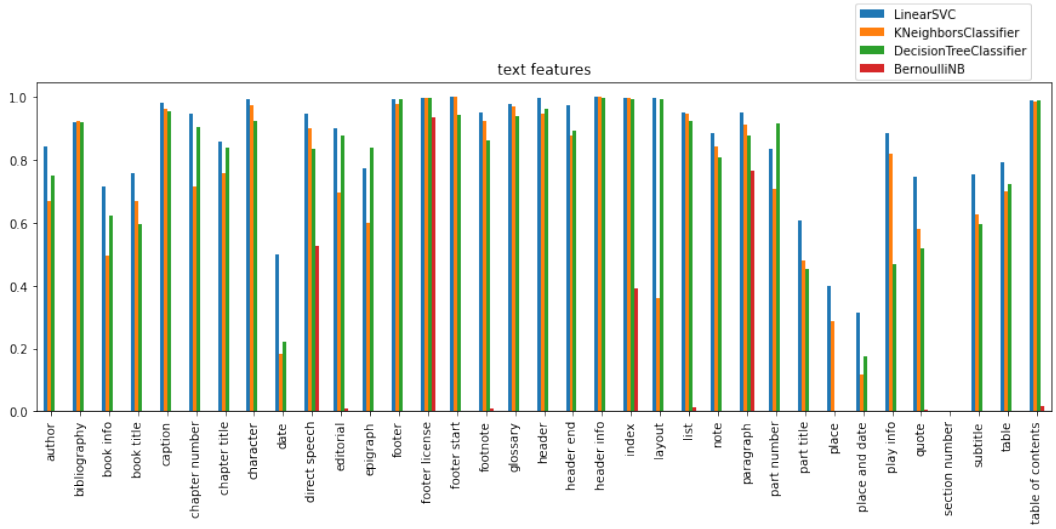


Figure 3 F1 measure for textual features.

Then, Boolean features (see Figure 4) do not perform well on the majority of classes. Those features were developed primarily to identify specific parts of the text.

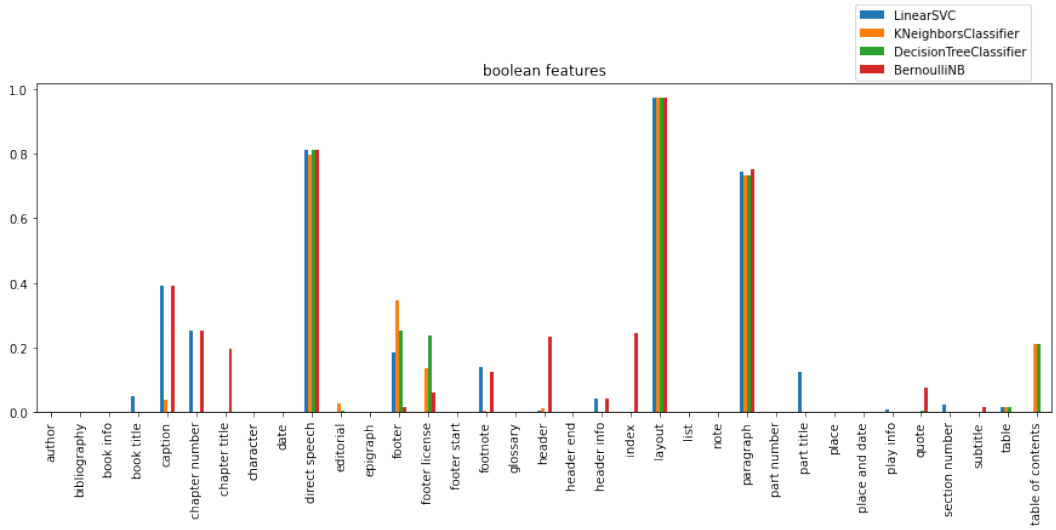


Figure 4 F1 measure for Boolean features.

The feature *Has asterisk* was meant to identify the *layout*, as there are almost always asterisks. According to the table, it predicts a *layout* category with an accuracy of 98%. Similarly, the Boolean feature *Has quote* is effective to identifying *direct speeches* thanks to the presence of quotation marks. Other Boolean features were not able to predict other classes. This is the case for the feature *Has bracket*, which was meant to identify *footnotes* and *captions*, as these parts are almost always contained between brackets in Gutenberg texts.

Like boolean features, numerical features (see Figure 5) fail to predict the majority of text parts. Interestingly, as far as numerical features are concerned, they have a different effect depending on the algorithm used. In fact, they seem to perform better with the DecisionTree algorithm than the others. Just as the BernoulliNB algorithm seems to outperform with the Boolean features.

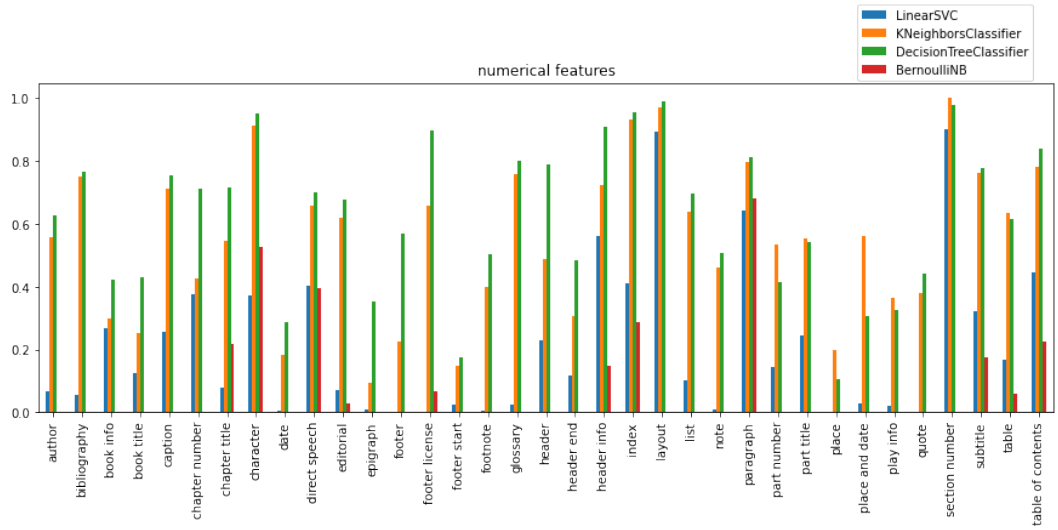


Figure 5 F1 measure for numeric features.

It is interesting to note that parts such as *direct speeches* or *quotes*, which in principle are similar in spelling, achieve results with a high percentage deviation due most likely to the lower number of supports for quotations.

However the general system shows very good results reaching scores above 90% for many classes. In particular, looking more closely at Figure 2, we can observe that some specific parts such as *captions*, *numbers* and *titles* of the chapter, as well as the *direct speeches* and *footers* achieve results above 90%.

4 Conclusion and Future Work

This paper presents a system for the automatic identification of parts of literary texts in the Gutenberg repository. Its aim is to provide scientists in the field of humanities with a tool to ease and fasten the access to textual analysis by identifying the narrative parts that are relevant to the textual analysis. With an overall accuracy of 93%, the system offers satisfying results.

The best performing features are the textual ones, which succeed in predicting almost all classes. Boolean and numerical features did not have a major influence on the classification, but help to identify specific parts of text. The two most recurrent classes, *direct speech* and *paragraph*, have been identified with a degree of precision of 95%. This high precision score is an encouraging result, as these two classes are the most relevant parts for textual analysis in literature.

In the future, further attention will be given to textual features. It would be interesting to explore these results further, by adding new textual features in order to improve the overall classification accuracy. In addition, we are planning to implement a systematic comparison between different classification algorithms. Our aim is to explore thoroughly the influence of each text feature in order to gain a better comprehension of the phenomenon.

References

1. Mohamed Aly. Survey on multiclass classification methods. *Neural Netw*, 19:1–9, 2005.
2. Ngo Xuan Bach, Nguyen Le Minh, Tran Thi Oanh, and Akira Shimazu. Learning logical structures of paragraphs in legal articles. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 20–28, Chiang Mai, Thailand, 2011. Asian Federation of Natural Language Processing. URL: https://www.aclweb.org/anthology/I11-1003.
3. Julian Brooke, Adam Hammond, and Graeme Hirst. GutenTag: an NLP-driven Tool for Digital Humanities Research in the Project Gutenberg Corpus. In *Proceedings of the Fourth Workshop on Computational Linguistics for Literature*, pages 42–47, 2015. doi:10.3115/v1/w15-0705.
4. R Bucher. *Classification of Fiction Genres: Text classification of fiction texts from Project Gutenberg*. diva-portal.org, 2018.
5. Hervé Déjean and Jean Luc Meunier. On tables of contents and how to recognize them. *International Journal on Document Analysis and Recognition*, 2009. doi:10.1007/s10032-009-0078-8.
6. Antoine Doucet, Gabriella Kazai, Bodin Dresevic, Aleksandar Uzelac, Bogdan Radakovic, and Nikola Todic. ICDAR 2009 book structure extraction competition. In *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2009. doi:10.1109/ICDAR.2009.280.
7. Antoine Doucet, Gabriella Kazai, and Jean Luc Meunier. ICDAR 2011 book structure extraction competition. In *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2011. doi:10.1109/ICDAR.2011.298.
8. Mattia Egloff and Davide Picca. The Project Gutenberg Ontology. In *European Association for Digital Humanities (EADH)*, Galway, Ireland, 2018.
9. Mattia Egloff, Davide Picca, and Alessandro Adamou. Extraction of character profiles from the gutenberg archive. In Emmanouel Garoufallou, Francesca Fallucchi, and Ernesto William De Luca, editors, *Metadata and Semantic Research*, pages 367–372, Cham, 2019. Springer International Publishing.
10. Liangcai Gao, Zhi Tang, Xiaofan Lin, Xin Tao, and Yimin Chu. Analysis of book documents' table of content based on clustering. In *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2009. doi:10.1109/ICDAR.2009.143.
11. M Gerlach and F Font-Clos. A standardized project gutenberg corpus for statistical analysis of natural language and quantitative linguistics. *Entropy*, 2020.
12. OL Goodloe. *Applications of Deep Neural Networks to Neurocognitive Poetics: A Quantitative Study of the Project Gutenberg English Poetry Corpus*. repository.asu.edu, 2019.
13. Shesen Guo, Ganzhou Zhang, Run Zhai, and Zehua Song. Distribution of English syllables in e-books of Project Gutenberg and the evolution of syllable number in two subcorpora. *Digital Scholarship in the Humanities*, 30(3):344–353, 2015. doi:10.1093/llc/fqu013.
14. Arthur M. Jacobs. The Gutenberg English Poetry Corpus: Exemplary Quantitative Narrative Analyses. *Frontiers in Digital Humanities*, 5, 2018. doi:10.3389/fdigh.2018.00005.
15. Gabriella Kazai, Antoine Doucet, Marijn Koolen, and Monica Landoni. Overview of the INEX 2009 book track. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2010. doi:10.1007/978-3-642-14556-8_16.
16. Evgeny Kim and Roman Klinger. Who feels what and why? annotation of a literature corpus with semantic roles of emotions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1345–1359, Santa Fe, New Mexico, USA, 2018. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/C18-1114.
17. Jiwei Li, Thang Luong, and Dan Jurafsky. A hierarchical neural autoencoder for paragraphs and documents. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1106–1115, Beijing, China, July 2015. Association for Computational Linguistics. doi:10.3115/v1/P15-1107.

18 Lara McConnaughey, Jennifer Dai, and David Bamman. The labeled segmentation of printed books. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 737–747, Copenhagen, Denmark, 2017. Association for Computational Linguistics. doi:10.18653/v1/D17-1077.

19 Davide Picca and Mattia Egloff. DHTK: The Digital Humanities ToolKit. In *Workshop on Humanities in the Semantic Web – WHiSe II*, pages 1–6, 2017.

20 Caroline Sporleder and Mirella Lapata. Automatic Paragraph Identification: A Study across Languages and Domains. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 72–79, 2004.

21 Joseph Worsham and Jugal Kalita. Genre identification and the compositional effect of genre in literature. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1963–1973, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/C18-1167.

22 Zhaohui Wu, Prasenjit Mitra, and C. Lee Giles. Table of contents recognition and extraction for heterogeneous book documents. In *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2013. doi:10.1109/ICDAR.2013.244.

A Data Augmentation Approach for Sign-Language-To-Text Translation In-The-Wild

Fabrizio Nunnari
German Research Center for Artificial Intelligence (DFKI), Saarland Informatics Campus D3.2, Saarbrücken, Germany

Cristina España-Bonet
German Research Center for Artificial Intelligence (DFKI), Saarland Informatics Campus D3.2, Saarbrücken, Germany

Eleftherios Avramidis
German Research Center for Artificial Intelligence (DFKI), Berlin, Germany

Abstract

In this paper, we describe the current main approaches to sign language translation which use deep neural networks with videos as input and text as output. We highlight that, under our point of view, their main weakness is the lack of generalization in daily life contexts. Our goal is to build a state-of-the-art system for the automatic interpretation of sign language in unpredictable video framing conditions. Our main contribution is the shift from image features to landmark positions in order to diminish the size of the input data and facilitate the combination of data augmentation techniques for landmarks. We describe the set of hypotheses to build such a system and the list of experiments that will lead us to their verification.

2012 ACM Subject Classification Computing methodologies → Machine learning; Human-centered computing → Accessibility technologies; Computing methodologies → Computer graphics

Keywords and phrases sing language, video recognition, end-to-end translation, data augmentation

Digital Object Identifier 10.4230/OASIcs.LDK.2021.36

Funding The research reported in this paper was supported by the BMBF (German Federal Ministry of Education and Research) in the project SOCIALWEAR (Socially Interactive Smart Fashion, DFKI Kst 22132).

1 Introduction

During the last years (multimodal) language technology has seen immense progress due to the great performance of deep neural networks working on large amounts of text, image or video data [26, 7, 1, 16, 15, 21]. This progress has enabled solutions and products which serve the majority of the consumer basis, which has the ability to speak and hear, but has comparatively neglected a considerable part of the population which is deaf or hearing impaired. In our effort to intensify research towards supporting deaf people with tools for their integration in the society, we focus on Sign Language (SL).

Sign Language is the main communication language for deaf people and used by more than 10 million people in the world [8]. People who are deaf from birth, also due to the lack of exposure to corresponding vocal signals, are not proficient in reading text translations and not comfortable with writing, as the spoken language is for them a foreign language. Hence, the only effective mean of communication are motion videos, either captured or played-back.

The research community has been investigating the machine-driven translation of SL for more than 20 years; at the beginning, with the introduction of text-to-SL tools to render sign language videos through the use of virtual characters [9, 12, 17]. More recently, the focus moved towards the more challenging SL video-to-text direction [25, 3], requiring a

Figure 1 An illustration of a possible application scenario: a hearing person, wearing a technologically augmented jacket embedding a camera, can follow the discussion between two sign language speakers. *Illustration by Mia Grote.*

more computational intensive analysis of video streams.

Within the SocialWear project,[1] we are conceiving solutions supporting an easier integration between the deaf and the speaking communities. One of the scenarios considered within the project aims at supporting the integration of a speaking person within a group of deaf speakers by translating, in real-time, sign language videos taken from wearable cameras, into voice (see Figure 1). The most challenging technological aspect being the need to recognize motion of people with diverse body proportions and clothing, framed from cameras positioned at different height, unpredictable framing angles, various lighting conditions and any background.

On the other hand, systems to estimate body and facial configurations "in the wild" do exist, see for instance [2, 5, 18], and could be trained due to the availability of big datasets. However, such training data for sign language translation of many national sign languages is missing (and will likely be missing for many years ahead), thus preventing the development of end-to-end translation systems in uncontrolled scenarios.

Therefore, we are proposing an approach to recognize sign language in the wild through a training pipeline that includes an augmentation of sign language animation data via synthetic generation (see Figure 2). The main idea is to delegate the identification of 3D landmarks in sign language video streams to specialized software, which is trained on big corpora in diverse conditions. Then, a 3D software will augment the 3D landmarks corpus to simulate cameras with different lenses and framing angles. Finally, train a neural network able to translate 3D landmark information into text, also bypassing any intermediate symbolic representation of sign language.

After introducing some related work (Section 2), in Section 3, we describe our envisioned pipeline and how to implement it. Section 4 describes our hypotheses and the evaluation plan, and finally Section 5 concludes the paper.

[1] `https://www.dfki.de/en/web/research/projects-and-publications/projects-overview/projekt/socialwear/`

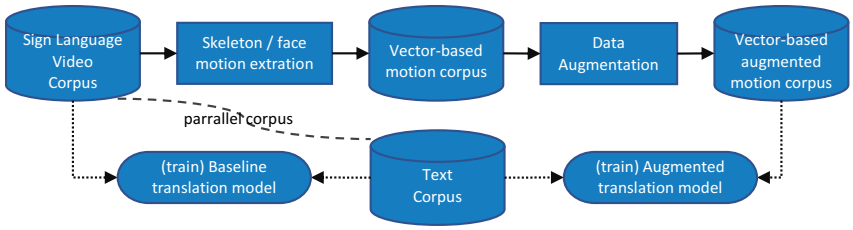


Figure 2 A diagram of the proposed translation pipeline.

2 Background

Very recent works approach SL video-to-text as a translation task that in most cases uses intermediate sign glosses [3, 4, 27, 28, 22]. These works employ a variety of neural machine translation (NMT) architectures, which mainly differ on how the input (video) is encoded – CNN, STMC networks, etc. Traditionally, the 2-steps conversion video-to-gloss followed by gloss-to-text has performed better than the end-to-end task. However, as currently happens in several deep learning problems, the use of transformer architectures [26] starts favouring end-to-end learning. Camgoz et al. (2020) [4] and Yin et al. (2020) [27] achieve state-of-the-art results on the RWTH-PHOENIX14T corpus [10, 11] – a standard test set for SL interpretation – with transformer-based NMT systems.

As already introduced, recognizing sing language motion in the wild would require an amount of data that is not available as of today. To circumvent this problem, we propose splitting the translation pipeline into a first phase, recognizing 3D landmarks from videos. This would allow augmenting the data by applying transformation techniques (e.g. simulating various recording conditions, size of body parts etc.) or creating additional features given the landmarks. Then the augmented vector-based information can be used to train the translation from landmarks into text.

Within this approach lies the work done by Ko et al. [14, 13]. They use NMT architectures as in previous works, but they extract 2D coordinates of human keypoints from the input videos and use these coordinates to train the neural translation systems. Vector-based animation data allows for applying object 2D normalization whereas the authors apply random frame skip sampling to augment the video data. Unfortunately, [4, 27] and [13] cannot be compared directly because systems are applied to different sign languages, domains, and test sets. Contrary to our suggestion, Ko et al. apply augmentation techniques directly on the video frames (prior to the landmark recognition) but not at the recognized landmarks (after the landmark recognition and before the translation from landmarks to text), as suggested by us.

More elaborated synthetic data augmentation techniques have been already employed in the generation of synthetic data for the task of recognizing hand poses. For example, Malik et al. [20, 19] generated more than 5 million images of hand configurations by setting up a virtual human in front of a desktop environment and simulating the random movement of a hand in front of a webcam. Also, Mueller et al. [23] generated a synthetic dataset by first capturing the motion of real hands, retargeting the motion to a virtual hand, framing it from an egocentric point-of-view, and then augmenting the dataset by modulating hand shape and skin color, adding occluding objects, and imposing random real-world backgrounds. In Covre et al. [6], the authors recorded gestures executed by a single human and transferred it to a virtual human. Then, they augmented the motion of the virtual character in order to train a gesture classification model based on random forests. The augmentation concerned both

modulating the gesture dynamics and moving the virtual camera. These techniques have not been applied to sign language, which is based on the movement of more body elements than just hands (e.g., posture, face).

3 Proposed methodology

State-of-the-art work from Camgoz et al. [4], achieves a BLEU score of more than 20 points by employing deep learning on an end-to-end translation approach from video to text. The input is processed within the neural network via a spatio-temporal embedding bound with a positional encoding. A major weakness of this approach is that the translation system relies on the raw input of a stream of video pixels, thus leading, from our point of view, to the following limitations:

- The resolution of the input video stream is forced by the architecture. Hence, videos at higher resolution must be scaled down even when a higher resolution, and thus more details, would be available. However, the higher the resolution of the video, the higher the computational power needed to train and run the neural network;
- The system is bound to the recording conditions of the corpus (RWTH-PHOENIX14T [10]) used for the experiments such as camera lenses (aperture and distortion), camera distance and angle, lighting conditions, background;
- The system is bound to the physical characteristics and the dress-code directives (black long-sleeved sweaters) present in the training corpus.

We assume that such a system would not be able to reach the same performance when tested on a video of a person with different clothing and skin colour, different light conditions or viewing distance or angle. This can be attributed to a known limitation of neural networks, which badly generalize for input coming from a different *distribution* than the one used for training and testing. An end-to-end neural architecture would be in principle able to learn to generalize the translation given different conditions, but this would require a vast amount of video-to-text parallel corpora, where the same phrases would be repeated under these different conditions. Unfortunately, the lack of sign language parallel corpora and the difficulty to obtain them is a major obstacle to building such a robust system.

Hence, we propose adding an intermediate level of indirection in the translation pipeline, by first extracting *motion data* of the SL speakers in terms of skeletal motion (for body and hands) and displacement of key points of the skin (for the face) from the video streams. Figure 2 shows a diagram of the proposed architecture. This way, the translation of the sign language into text is performed on the animation data of a 3D virtual human, rather than on the video of a real human. This would lead to several advantages:

1. There are sufficient data and very strong existing models for recognizing skeletal and facial motion in-the-wild, such as OpenFace [2], OpenPose [5], and MediaPipe [18];
2. the performance of the system would not be affected by the identity of the signer nor by their clothes;
3. the translation system would be independent from lighting conditions;
4. it would be possible to provide the network with additional data points, like the distances between joints (a feature often very useful in hand pose or gesture recognition);
5. the size of the skeleton and the face could be normalized to improve the sign recognition on bodies with very different proportions;
6. it opens the possibility to augment the animation information through the simulation of different camera position and lenses via geometrical transformations, thus training a system able to recognize signs from different distances and shooting angles;

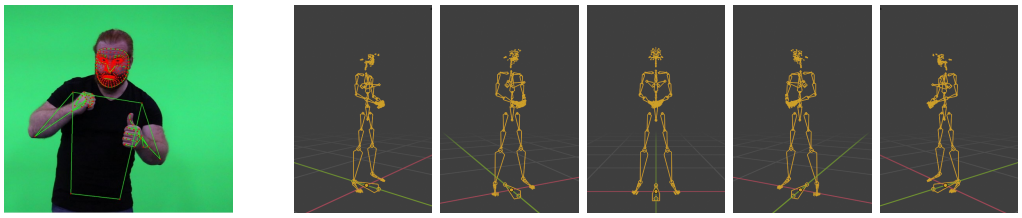


Figure 3 Preliminary work on (left) capturing the skeletal motion from a user and (right) augmenting skeletal motion as seen from multiple points of view [24].

7. the neural architecture dedicated to the translation of motion data into text would be smaller, and hence faster and more energy efficient, as the quantity of information received as input would be two orders of magnitude inferior to video data.

Concerning the last point, as a rough estimation of the reduction in the quantity of information, consider that one second of RGB color video at resolution 210x260 pixels (as for RWTH-PHOENIX14T) at 25 FPS would require 4095 KBytes. In contrast, animation data, counting roughly 60 bones for the upper body (4-tuple quaternions for the rotations) and 468 face landmarks (3-tuple for each vertex in space), encoded as 4-byte floats, makes 6576 bytes per frame, which recorded at 25 FPS leads to approximately 164 KBytes per second. This is about 4% of the corresponding low-resolution video data.

Points 4-7 above demonstrate an advantage as compared to state-of-the-art Ko et al. [14, 13], as the augmentation will occur directly on the landmarks, providing additional data related to observed weaknesses of the existing models.

A drawback of this approach is that any error introduced by the skeletal and the facial recognition stage would propagate to the translation stage. Nevertheless, given the consistent improvement of the technologies specifically dedicated to the motion tracking of body, hands, and face we are confident that the tracking errors introduced by the motion analysis stage would be limited and well compensated by the advantages of a lighter architecture dedicated to the translation process. Additionally, if deemed necessary, deep learning offers the possibility of handling both the skeletal/face recognition and the translation through the same joint neural network, which would minimize the effects of error propagation.

4 Empirical Evaluation Plan

As already introduced, we plan to extract skeletal and facial motion data from videos, and use those to feed a MotionData-to-text (MD2Text) translation system. For the extraction of motion data, we plan to use the recent MediaPipe[2] [18] framework, which provides tools for the extraction of body, hands, and facial motion data. Figure 3 shows the result of initial tests. As for the corpus, we will use the RWTH-PHOENIX-14T dataset as it has been used in previous related research such as that of Camgoz et al. [4].

We can summarize our experiments with the following pipeline:

1. Retrieve the corpus `V` of videos from RWTH-PHOENIX-14T;
2. create a baseline model `V2Text`, which takes plain videos as input: train it and measure its performances on corpus `V`;

[2] `https://mediapipe.dev/`

3. create a corpus `MD` (motion data for body, hands, and face) by analysing the videos of `V` using MediaPipe;
4. define a model `MD2Text`, which takes motion data as input: train it and measure its performances on corpus `MD`;
5. augment the `MD` corpus with additional feature like mutual distances between joints (`MD+D`), camera settings and positions (`MD+C`), and by normalizing body proportions (`MD+B`);
6. train a model `MD+2Text` on the augmented `MD+D+C+B` corpus and measure its performances;
7. create a corpus of "videos in the wild" (`WV`) of new signers, with random clothes, diverse camera framing angles and lenses;
8. measure the performances of `V2Text` on `WV`;
9. extract the motion data `WMD` from `WV`; and
10. measure the performances of `MD2Text` and `MD+2Text` on `WMD`.

The goal of the set of experiments is to verify the following hypotheses:

- **H1**: `V2Text` performs worse on `WV` than on `V`;
- **H2**: `MD2Text` performs better than `V2Text`;
- **H3**: `MD2Text` and `MD+2Text` require much less computational resources than `V2Text` for both training and inference (for the latter, sum up the inference time of MediaPipe);
- **H4**: `MD+2Text` performs better than `MD2Text` when tested on `MD`;
- **H5**: `MD+2Text` performs better than `MD2Text` when tested on `WMD`;
- **H6**: finally, `MD+2Text` performs on `WMD` as good as on the original `MD`.

5 Summary

In this paper we have described and analyzed the current main approaches for the translation of sign language into text, and detected the main weaknesses for an application in real daily life. To overcome those limitations, we proposed an approach based on chaining a video-to-motion recognition system followed by an end-to-end translation approach from motion vectors into text.

Whereas, nowadays, researchers are proving the superiority of pure end-to-end architectures trained on huge quantities of "dirty" data, such approach cannot be yet applied in the context of sign language because of the scarcity of resources. In general, in this work we are exploring the possibility of training systems starting from a smaller quantity of "clean" data (recorded in controlled conditions) and improve performances through an artificial introduction of data variability. An alternative and complementary approach would be the exploitation of fine-tuning and domain adaptation techniques which would allow using models trained in richer settings (such as video captioning, video question answering or video and language inference) as initialisation of sign language translators. This is another possible research line left as future work and has been not considered in the discussion.

A reasonable limitation of our approach is that data augmentation is not really equivalent to increasing the sampling size, but rather a localized exploration of the neighbourhood of existing samples along some of the features characterizing the input domain. Still, data augmentation has proven to be effective in image classification, and it is part of the challenge to prove that it will be effective on motion analysis too.

In the presented proposal, we described the idea of augmenting the data mainly by generating different camera framing conditions. In future work, we could explore the benefits of augmentation applied to the human motion, too, by performing a modulation of the dynamics of the motion (e.g., time scaling and time warping) and by the manipulation of motion trajectories.

References

1 Chris Alberti, Jeffrey Ling, Michael Collins, and David Reitter. Fusion of detected objects in text for visual question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2131–2140, Hong Kong, China, November 2019. Association for Computational Linguistics. doi:10.18653/v1/D19-1219.

2 Tadas Baltrusaitis, Amir Zadeh, Yao Chong Lim, and Louis-Philippe Morency. OpenFace 2.0: Facial Behavior Analysis Toolkit. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 59–66, Xi'an, 2018. IEEE.

3 Necati Cihan Camgöz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. Neural sign language translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7784–7793, 2018. doi:10.1109/CVPR.2018.00812.

4 Necati Cihan Camgöz, Oscar Koller, Simon Hadfield, and Richard Bowden. Sign language transformers: Joint end-to-end sign language recognition and translation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10020–10030. IEEE, 2020. doi:10.1109/CVPR42600.2020.01004.

5 Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *IEEE transactions on pattern analysis and machine intelligence*, 43(1):172–186, 2019.

6 Nicola Covre, Fabrizio Nunnari, Alberto Fornaser, and Mariolino De Cecco. Generation of action recognition training data through rotoscoping and augmentation of synthetic animations. In *Augmented Reality, Virtual Reality, and Computer Graphics*, pages 23–42, Cham, June 2019. Springer International Publishing. doi:10.1007/978-3-030-25999-0_3.

7 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi:10.18653/v1/N19-1423.

8 David M. Eberhard, Gary F. Simons, and Charles D. Fenning. *Ethnologue: Languages of the World. Twenty-third edition.* SIL International, Dallas, Texas, 2020.

9 R. Elliott, J. R. W. Glauert, J. R. Kennaway, and I. Marshall. The development of language processing support for the visicast project. In *Proceedings of the Fourth International ACM Conference on Assistive Technologies*, Assets '00, page 101–108, New York, NY, USA, 2000. Association for Computing Machinery. doi:10.1145/354324.354349.

10 Jens Forster, Christoph Schmidt, Thomas Hoyoux, Oscar Koller, Uwe Zelle, Justus Piater, and Hermann Ney. RWTH-PHOENIX-Weather: A Large Vocabulary Sign Language Recognition and Translation Corpus. In *Language Resources and Evaluation*, pages 3785–3789, Istanbul, Turkey, May 2012.

11 Jens Forster, Christoph Schmidt, Oscar Koller, Martin Bellgardt, and Hermann Ney. Extensions of the Sign Language Recognition and Translation Corpus RWTH-PHOENIX-Weather. In *LREC*, pages 1911–1916, 2014.

12 Alexis Heloir and Michael Kipp. EMBR - A Realtime Animation Engine for Interactive Embodied Agents. In *Proceedings of the 9th International Conference on Intelligent Virtual Agents (IVA-09)*, 2009.

13 Sang-Ki Ko, Kim Kim, Hyedong Jung, and Choong sang Cho. Neural sign language translation based on human keypoint estimation. *Applied Sciences*, 9:2683, 2019.

14 Sang-Ki Ko, Jae Gi Son, and Hyedong Jung. Sign language recognition with recurrent neural network using human keypoint detection. In *Proceedings of the 2018 Conference on Research in Adaptive and Convergent Systems*, RACS '18, page 326–328, New York, NY, USA, 2018. Association for Computing Machinery. doi:10.1145/3264746.3264805.

15 Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. HERO: Hierarchical encoder for Video+Language omni-representation pre-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2046–2065, Online, November 2020. Association for Computational Linguistics.

16 Xiujun Li, Xi Yin, Chunyuan Li, Xiaowei Hu, Pengchuan Zhang, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-semantics aligned pre-training for vision-language tasks. *arXiv preprint*, 2020. arXiv:2004.06165.

17 Vincenzo Lombardo, Cristina Battaglino, Rossana Damiano, and Fabrizio Nunnari. An avatar-based interface for the italian sign language. In *Proceedings of the 2011 International Conference on Complex, Intelligent, and Software Intensive Systems*, CISIS '11, pages 589–594, Washington, DC, USA, June 2011. IEEE Computer Society. doi:10.1109/CISIS.2011.97.

18 Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. MediaPipe: A Framework for Building Perception Pipelines. *arXiv:1906.08172 [cs]*, 2019. arXiv:1906.08172.

19 Jameel Malik, Ahmed Elhayek, Fabrizio Nunnari, and Didier Stricker. Simple and effective deep hand shape and pose regression from a single depth image. *Computers & Graphics*, 85:85–91, 2019. doi:10.1016/j.cag.2019.10.002.

20 Jameel Malik, Ahmed Elhayek, Fabrizio Nunnari, Kiran Varanasi, Kiarash Tamaddon, Alexis Heloir, and Didier Stricker. Deephps: End-to-end estimation of 3d hand pose and shape by learning from synthetic depth. In *2018 International Conference on 3D Vision (3DV)*, pages 110–119. IEEE, September 2018. doi:10.1109/3DV.2018.00023.

21 Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

22 Taro Miyazaki, Yusuke Morita, and Masanori Sano. Machine translation from spoken language to sign language using pre-trained language model as encoder. In *Proceedings of the LREC2020 9th Workshop on the Representation and Processing of Sign Languages: Sign Language Resources in the Service of the Language Community, Technological Challenges and Application Perspectives*, pages 139–144, Marseille, France, May 2020. European Language Resources Association (ELRA).

23 Franziska Mueller, Dushyant Mehta, Oleksandr Sotnychenko, Srinath Sridhar, Dan Casas, and Christian Theobalt. Real-time hand tracking under occlusion from an egocentric rgb-d sensor. In *Proceedings of the IEEE ICCV Workshops*, October 2017.

24 Florian Schicktanz, Lan Thao Nguyen, Aeneas Stankowski, and Eleftherios Avramidis. Evaluating the translation of speech to virtually-performed sign language on AR glasses. In IEEE, editor, *Proceedings of the Thirteenth International Conference on Quality of Multimedia Experience (QoMEX)*, 2021.

25 Stephanie Stoll, Necati Cihan Camgoz, Simon Hadfield, and Richard Bowden. Sign Language Production using Neural Machine Translation and Generative Adversarial Networks. In *British Machine Vision Conference*, Northumbria, UK, 2018. British Machine Vision Association.

26 Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008. Curran Associates, Inc., 2017.

27 Kayo Yin and Jesse Read. Better sign language translation with STMC-transformer. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5975–5989, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. doi:10.18653/v1/2020.coling-main.525.

28 Jiangbin Zheng, Zheng Zhao, Min Chen, Jing Chen, Chong Wu, Yidong Chen, Xiaodong Shi, and Yiqi Tong. An Improved Sign Language Translation Model with Explainable Adaptations for Processing Long Sign Sentences. *Computational Intelligence and Neuroscience*, 2020:11, 2020. doi:10.1155/2020/8816125.

A Review and Cluster Analysis of German Polarity Resources for Sentiment Analysis

Bettina M. J. Kern
University of Vienna, Austria

Andreas Baumann
University of Vienna, Austria

Thomas E. Kolb
TU Wien, Austria

Katharina Sekanina
University of Vienna, Austria

Klaus Hofmann
University of Vienna, Austria

Tanja Wissik
Austrian Academy of Sciences, Vienna, Austria

Julia Neidhardt
TU Wien, Austria

Abstract

The domain of German polarity dictionaries is heterogeneous with many small dictionaries created for different purposes and using different methods. This paper aims to map out the landscape of freely available German polarity dictionaries by clustering them to uncover similarities and shared features. We find that, although most dictionaries seem to agree in their assessment of a word's sentiment, subsets of them form groups of interrelated dictionaries. These dependencies are in most cases an immediate reflex of how these dictionaries were designed and compiled. As a consequence, we argue that sentiment evaluation should be based on multiple and diverse sentiment resources in order to avoid error propagation and amplification of potential biases.

2012 ACM Subject Classification Computing methodologies → Cluster analysis

Keywords and phrases cluster analysis, sentiment polarity, sentiment analysis, German, review

Digital Object Identifier 10.4230/OASIcs.LDK.2021.37

Supplementary Material *Software (Source Code)*: `https://github.com/bettina-mj-kern/LDK_2021`

Funding This research was funded by the City of Vienna (Digitaler Humanismus grant, MA7-737909/19).

1 Introduction

Sentiment analysis is a popular tool to draw emotional information from language data. One approach for sentiment detection is the use of lexical resources, i.e., sentiment dictionaries containing a list of words and their corresponding sentiment information. An ample selection of sentiment dictionaries exists for the English language. German, however, with its abundance of compound words, inflections and derivational suffixes, poses more of a challenge for automated sentiment analysis [6] and for the development of adequate tools and methods.

Many sentiment dictionaries contain sentiment ratings on more than one aspect than just polarity. The dimensional view is a common conception of emotion, in which emotions are characterised as quantitatively different from each other on a number of dimensions [8, 24, 34]. Accordingly, different dimensions can be used to describe the sentiment information of a word, polarity being one of them. Different sentiment dictionaries make use of different conceptualisations of emotions and include different dimensions, such as arousal, valence, or dominance, to capture the emotional content of words. This makes them difficult to compare. For this paper, we thus focus on the common denominator for most German sentiment resources: sentiment polarity, also often referred to as valence and sometimes as evaluation.

The assessment of polarity is one of the most basic tasks in sentiment classification and reflects whether a word or a text snippet is positive or negative. While the number and the definition of emotional dimensions may vary between dictionaries, most of them contain ratings on sentiment polarity. Dictionaries also differ in how they encode polarity: some of them provide categorical polarity labels (eg. NEG, POS), while others give numerical values which allow measuring sentiment intensity in addition to sentiment orientation. When it comes to the German language, sentiment resources are relatively small in size with regard to the number of words they encompass, mostly containing up to a few thousand words [18].

Although there are websites[1] listing different databases, data sets and sentiment dictionaries for German sentiment analysis, none of them are exhaustive. Currently, there is no central, comprehensive go-to online resource for German sentiment analysis.

Our paper aims to collect and compare the available resources and map out the landscape of German polarity dictionaries for sentiment analysis. To this end, we analyse the similarities between the 15 sentiment dictionaries identified by our search by means of a divisive clustering method.

The results show that while most dictionaries seem to make relatively comparable predictions about a word's sentiment, distinct subgroups can be identified, which are partially determined by how the dictionaries were developed. First, we find that dictionaries with categorical sentiment labels tend to behave similarly. More interestingly, however, we also identify groups of similar sentiment dictionaries that depend on each other by design. Based on our observations, we argue that newly compiled polarity dictionaries should be based on either diverse extant resources or newly created sentiment annotations to avoid error propagation and the amplification of potentially present biases.

Our paper is structured as follows. First, we describe the dictionaries analysed in this paper and the preprocessing steps that we applied. We then briefly describe the clustering approach applied to our data set and present the results of our analysis. Finally, we discuss and critically assess our findings.

2 Data and Preprocessing

For this research, we thoroughly combed the web for German-language polarity dictionaries that contain polarity ratings and identified 15 resources in total. The search was limited to already existing sentiment dictionaries that were freely available on the internet for academic and non-commercial purposes and contained a polarity rating of words. Databases containing annotated data that could theoretically be used to create a sentiment dictionary were not considered. The dictionaries vary considerably in their development processes and methods. Table 1 gives an overview over the identified resources.

With regard to the scope of this research, we focus on the dimension of polarity to include and analyse as many resources as possible, although we are aware of the relevance of a multidimensional approach to emotions, in particular in psychological research. Consequently, most of the sentiment resources included in our analysis contain more than one sentiment dimension. AffDict [36] and AffMeaning [2] contain sentiment ratings on potency (strong vs. weak) and activity (calm vs. lively) in addition to evaluation (good vs. bad, i.e. polarity). With these dimensions, AffDict and AffMeaning adhere to the dimensional view of Osgood, Suci and Tannenbaum with evaluation, potency and activity constituting an affective space in language [25]. AffNorms [18] includes four psycholinguistic attributes: abtractness/concreteness, arousal, imageability and valence (i.e. polarity). BAWL-R contains ratings on

[1] e.g. `https://sites.google.com/site/iggsahome`

imageability, arousal and valence, as well as linguistic properties of words that may influence their perception [40]. LANG [13] and Wordnorms [19] have ratings on valence, arousal and concreteness.

The authors who created the Morph resource [33] point out that in their experiments on sentiment classification using the full set of dimensions yields higher prediction accuracy than using just one dimension, but if one single dimension is used, polarity ratings are most predictive. Thus it seems that polarity can serve as a reasonable proxy in cases as ours where it is not possible to include the full range of dimensions in the analysis.

The dictionaries also vary considerably in the methods that were used to create them. In five cases, sentiment ratings were collected from human annotators: AffDict [36], AffMeaning [2], BAWL-R [40], LANG [13] and Wordnorms [19]. Other dictionaries rely on already existing resources. The Polarity Clues were created by translating existing English sentiment resources and enriching them with synonyms [41]. SentiMerge was created by simultaneously combining polarity scores from several extant sentiment dictionaries using a Bayesian probabilistic model [11]. AffNorms [18] used three already existing German sentiment resources as training data to automatically infer polarity values for over 350 000 words using a supervised machine learning algorithm following Turney et al. [37]. SentiWS [29] is based on automatically translated entries of the General Inquirer, a German collocation dictionary and collocation analysis, using pointwise mutual information (PMI) to assign polarity weights, following the approach by Turney and Littmann [38]. EmotionDict [16] uses already existing German sentiment resources (among them the Polarity Clues [41] and SentiWS [29]), enriching them with synonyms. SePL [31] was created by extracting opinion-bearing phrases of reviews and using the star ratings to infer opinion values. Morph [33] used the Polart lexicon [15] and added words from various databases to infer the sentiment values of German compound words based on their morphological structure.

ANGST pursues a mixed strategy by using the valence, arousal and imageability ratings of the BAWL-R [40], and supplementing them with ratings in dimensions for additional words and with an additional dimension, dominance.

For the ALPIN dictionary, human annotators rated text snippets from the Austrian Media Corpus [28] as positive or negative in a crowd-sourcing survey. The sentiment value of a word was then determined by the number of negative and positive texts it appears in, as proposed by [1].

As Table 1 shows there is a lot of heterogeneity among the dictionaries. Basic preprocessing steps were applied to normalise the dictionaries in order to merge, analyse and compare them. The steps of the preprocessing depend on the specific dictionary's structure. For a detailed account of the preprocessing applied to each dictionary, refer to Appendix A.

The sentiment values in AffDict [36], AffMeaning [2], AffNorms [18], ANGST [35], BAWL-R [40], LANG [13], SentiMerge [11] and Wordnorms [19] had to be rescaled to the interval [−1,1], i.e., going from maximally negative to maximally positive.

Table 1 Overview of the 15 German polarity resources included in this analysis.

Label	Reference	Description	Method	Size	Scale	Purpose
AffDict	Schröder, 2011 [36]	3 dimensions: Evaluation (good–bad), potency (strong–weak), activity (lively–calm)	1 905 human raters	1 100 total, 376 nouns, 393 verbs, 331 adjectives, 128 combinations	[−4,4]	Modelling impression formation
Aff-Meaning	Ambrasat et al., 2014 [2]	3 dimensions: Evaluation (good–bad), potency (strong–weak), activity (lively–calm); concepts related to authority and community	2 849 human raters	909 words from semantic fields of authority and community	averaged 9-point semantic differential scale	investigate intra-societal consensus and variation in affective meanings
AffNorms	Köper & Schulte im Walde, 2016 [18]	Ratings on 4 psycho-linguistic affective attributes	Supervised Machine Learning	350 000 nouns, verbs and adjectives	[0,10]	Ratings for sentiment analysis
ALPIN	Kolb et al., 2021	Austrian words from the political and media context	Machine learning on human-rated text snippets	3 636 nouns, verbs and adjectives	[−1,1]	Sentiment analysis on Austrian words
ANGST	Schmidtke et al., 2014 [35]	Words of the ANEW rated on 6 dimensions	human raters	1 003 nouns, verbs and adjectives	[−3,3]	Affective word lists for experiments
BAWL-R	Võ et al., 2009 [40]	Extension of BAWL with 3 psycholinguistic indices	200 human raters	>2 900	[−3,3]	Affective word lists for experiments
Emotion-Dict	Klinger, Suliya & Reiter, 2016 [16]	Sentiment labels for 7 basic emotions	3 human raters	4101 nouns, verbs and adjectives	POS, NEG	Emotions detection in literary texts
Polarity Clues	Waltinger, 2010 [41]	Translated English polarity features	Translation of English sentiment resources	10 141 nouns, verbs and adjectives	NEG, NEU, POS	Sentiment analysis
LANG	Kanske & Kotz, 2010 [13]	Ratings for emotional valence, arousal and concreteness	64 human rater, 2 years apart	1 000 nouns	[0,10]	Acquire test-retest reliability for ratings

Table 1 continued from previous page

Label	Reference	Description	Method	Size	Scale	Purpose
Morph	Ruppenhofer & Wiegand, 2017 [33]	Sentiment ratings for rare and complex compounds	Human rated baseline, rule-based classifier	8 400 in total from several samples	NEG, NEU, POS	Approach the problem of coverage in German polarity resources
PolArt	Klenner, Fahrni & Petrakis, 2009 [15]	Lexical resource for German sentiment analysis	Semi-automatic translation approach	10 790	POS, NEU, NEG, SHI, INT	Ratings for sentiment analysis
SePL	Rill et al., 2012 [31]	Opinion bearing words and phrases for German	Infer sentiment value from review rating	Adjectives, nouns, as well as adjective and noun-based phrases	[−1,1]	Inclusion of intensifiers, reducers and negation words
SentiMerge	Emerson & Declerck, 2014 [11]	Combine polarity scores from several data sources and estimate the quality of each source	Normalising and merging with Bayesian framework	98 918 words from Polarity Clues, SentiWS and Polart	[−1.7,1.7]	Ratings for sentiment analysis
SentiWS	Remus, Quasthoff & Heyer, 2010 [29]	Affect dictionary with syntactic category, inflectional forms, polarity and strength	Semi-supervised Machine Learning: Pointwise Mutual Information (PMI)	3468 nouns, verbs, adverbs, adjectives, 1650 negative, 1 818 positive	[−1,1]	Ratings for sentiment analysis
WordNorms	Lahl et al., 2009 [19]	Ratings on 3 psycho-linguistic attributes: concreteness, valence, and arousal	3907 human raters, crowdsourced via webapp	2654 nouns	[0,10]	Ratings for sentiment analysis

The part-of-speech (PoS) tagging in the individual dictionaries is very inconsistent and not all dictionaries provide it. The largest dictionary with 350 000 entries does not provide PoS labels which means that for the vast majority of words our analysis, there is no part-of-speech information is available to begin with. PoS information is consequently not considered and was removed during preprocessing.

In cases of dictionaries with discrete categories ("negative", "positive", "neutral"), labels were replaced with numerical values to allow quantitative analyses on the dictionaries. To this end, the dictionaries with numerical sentiment values were merged first and two separate means were calculated for positive and negative values. The mean of the negative numerical sentiment values ($mean = -0.228$) was then imputed for words that were labeled "negative". The same was done, *mutatis mutandis*, for the words labeled "positive" ($mean = 0.176$).

Figure 1 shows the distribution of mean sentiment values for all words in the merged sentiment dictionary. As can be seen, most words have a sentiment value close to zero, indicating neutral polarity.

Most of the dictionaries range from a few thousand to ten thousand words, as is reflected by the median length of the dictionaries ($median = 3702$). Note that these values relate to the dictionaries *after* preprocessing and cleaning.

As the difference between the smallest with about a hundred and largest dictionary with over 350 000 words is considerable, most words are covered by only one dictionary. Consequently, the data set is relatively sparse. Only around 55 000 words appear in two or more dictionaries, and not a single word is included in all 15 dictionaries. Note that the sparsity of our data set is mainly a reflex of the AffNorms [18] being more than three times larger than the second largest, SentiMerge [11].

The preprocessed dictionaries were finally merged into a large master dictionary comprising 15 dictionaries and polarity information for roughly 400 000 words. The entries consist mostly of single words, but there are also entries that consist of more than one word, since one of the dictionaries, SePL [31] is composed of short phrases and adverb-adjective combinations.

3 Analysis

All analyses as well as the preprocessing were done using `R`, version 4.0.3 [27] and a selection of R packages. In order to compare all dictionaries, we adopted a clustering approach, using the `cluster` package, version 2.1.0 [21].

In order to cluster the dictionaries, the data were arranged in a matrix with 15 rows and roughly 400 000 columns. Next, a distance matrix was calculated based on Euclidean distance. As the sentiment values are already scaled to the interval $[-1,1]$ after the preprocessing, further standardization was not required. We opted for Euclidean distance since this distance measure (as any Minkovski-type distance) is more sensitive to distributional differences than, for example, correlation dissimilarity. Thus, we can more accurately compare and find differences between, for example, dictionaries with a relatively centered distribution of sentiment scores on the one hand and more dispersed dictionaries on the other hand. Importantly, for each pair of dictionaries the distance measure was only based on the set of overlapping words contained in both dictionaries.

Clustering is an unsupervised technique used to group objects which are close to each other in a multidimensional feature space to uncover inherent structures in the data [4]. Optimally, the objects in the same cluster show a high degree of similarity while being as dissimilar as possible from objects belonging to different clusters [14]. There are different algorithms to achieve this. One main distinction can be made between partitioning and

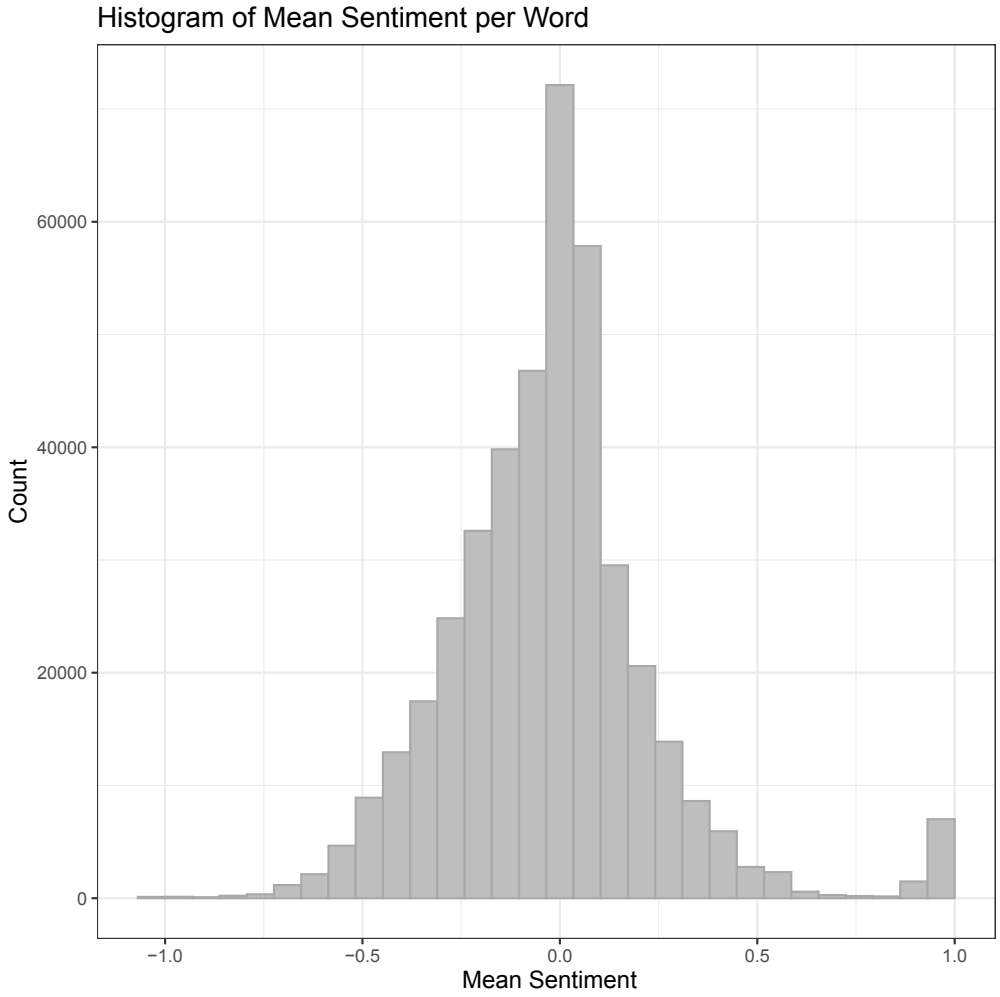


Figure 1 Histogram showing the mean sentiment value per word across all merged dictionaries.

hierarchical methods. Partitioning methods construct a predefined number of k clusters. Hierarchical methods do not construct a single partition with k clusters, but output the situation for $k = 1$ cluster to $k = n$ and all values of k in between [14].

There is no clear consensus about which algorithm is best [3] and cluster validation is a difficult task, as it lacks a common theoretical background and clear-cut best practices or rules [3]. Several cluster validation indices exist, but previous studies have shown that no single index is able to outperform the rest [7, 22, 23]. Further, the performance of the used evaluation criteria depends on the data [23]. All these factors make it difficult to determine the optimal parameters for the cluster analysis at hand.

To identify the parameters that work best for the given data and to assess cluster stability, different cluster algorithms and cluster definition methods were evaluated using the `clValid` package [4]. Cluster evaluation indices can be roughly categorized into external and internal measures. External validation measures rely on an outside data source with known class labels that serve as benchmark data [12]. Such a gold standard does not exist in many cases. In particular, there are no benchmarks concerning a word's "true" sentiment value. We thus relied on internal cluster validation measures that use the clustering and the underlying data set to assess the quality of the clustering. Three internal validation measures concerning the compactness, connectedness and separation of the clusters can be calculated with `clValid` [4].

The package provides a function to facilitate permutating different distance metrics and cluster methods, allowing to assess the robustness of the identified cluster solution. Rank Aggregation, as supported by `RankAggreg` package [26], was used to summarise the results in a super-list with the top three winning cluster algorithms plus optimal k, ranked by how much they maximise connectivity [4] and silhouette width [32] and minimise the Dunn Index [9].

In the present analysis, three different cluster algorithms are compared: Agglomerative nesting, partitioning around medoids and divisive analysis. The influence of different linkage methods is also evaluated, as well as different values of k from 1 to 5.

Divisive analysis is a hierarchical clustering method that starts out with a single cluster containing all objects and works bottom-up. In each step, the object that is most dissimilar to all other objects is identified and separated into a splinter group. All other objects are either assigned to the new splinter group or remain in their original cluster, depending on their similarity. In each iteration, the cluster with the largest diameter is selected and one object is separated until there are $k = n$ clusters.

Agglomerative nesting works on a reverse logic. At the beginning, each objects starts out as an individual cluster. In the first step, the two most similar objects are fused into one cluster. All distances are recalculated, and the process is repeated until all objects form a single, large cluster. An important parameter is the linkage method that determines the similarity between two objects. Several linkage methods exist, three commonly used ones are average linkage, complete linkage and Ward linkage. Complete linkage merges two clusters with the smallest maximum distance between them. Average linkage fuses clusters with the smallest average distance between them. Ward's method merges the two clusters that provide the smallest increase in within-cluster variance.

Partitioning around medoids clustering requires a predefined number k of clusters that the user wants to extract. The algorithm then selects k representative objects in the data. The clusters are formed by assigning each remaining object to the nearest representative object, the medoid [14]. The average distance (or dissimilarity) of the representative object to all objects of the same cluster is minimised. The principle is similar to k-means clustering which aims to minimise the average distance, making it susceptible to outliers. In this regard, partitioning around medoids is the more robust method.

The divisive analysis algorithm with three clusters yielded the best outcome for the three quality metrics. Consequently, we conducted divisive clustering with $k = 3$ clusters. A more detailed account of the employed methods and cluster evaluation can be found in the supplementary materials (`https://phaidra.univie.ac.at/o:1169856`).

4 Discussion

The cluster dendrogram reveals interesting insights. First of all, the cluster analysis suggests that most dictionaries are relatively similar to each other, as they form a single, large group in the dendrogram (left-most cluster in Figure 2). We will take a closer look at the internal structure of this group before we discuss the two dictionaries representing outliers (SePL and Polart on the right in Figure 2).

First, it is noteworthy that some dictionaries in the large cluster were created by extending or building on already existing ones. ANGST [35] uses the ratings of valence, arousal, and imageability of BAWL-R [40] as a basis and extends them with ratings on dominance and potency, additional words and new arousal ratings.

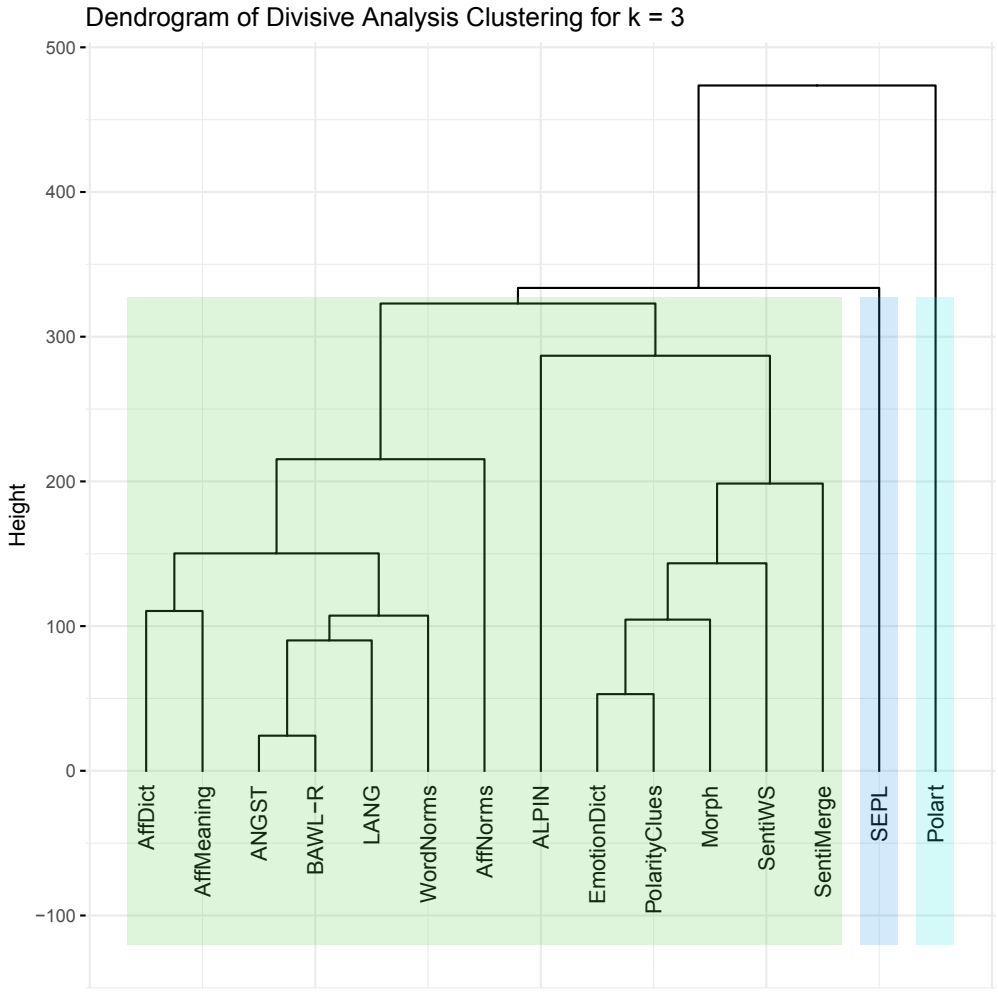


Figure 2 Dendrogram of divisive clustering using Euclidean distances. The $k = 3$ clusters are highlighted.

Thus, it is not surprising that these two dictionaries are highly similar to each other and were separated into different clusters only in the last iteration of the divisive algorithm.

Interestingly, ANGST [35], LANG [13] and BAWL-R [40] share a common methodological feature: they all used self-assessment manikins [20] for collecting the sentiment ratings. As can be seen in Figure 2, the three dictionaries have a relatively high similarity and were separated at a late step in the divisive clustering process. This highlights the role of the data collection procedure in compiling sentiment dictionaries.

In a similar way, there are dependencies between other dictionaries as well. AffDict [36] and AffMeaning [2] were not created for the purpose of conducting sentiment analyses, they were the result of two studies in social psychology. In both instances, survey participants were asked to rate words on the same three dimensions: on evaluation (good vs. bad), potency (strong vs. weak), and activity (lively vs. calm). AffDict [36] was used to model impression formation. AffMeaning [2] was used to examine intra-societal consensus and variation in affective meanings of concepts related to authority and community. The author of the AffDict paper [36] also collaborated on the paper on AffMeaning [2], the other authors on that paper

appear to be lab colleagues. While the research topics are different, the general methodology seems be rather similar and may explain why these two dictionaries ended up in the same cluster.

The very extensive AffNorms (with over 350 000 words) made use of BAWL-R [40], Wordnorms [19] and LANG [13] as training data for a supervised machine learning algorithm. This allows to automatically generate sentiment values for large amounts of words. AffNorms appears in the same cluster as its three seed dictionaries, but somewhat removed from them.

AffDict [36], AffMeaning [2], BAWL-R [40], LANG [13] and Wordnorms [19] are clustered quite closely to each other. As pointed out above, they share a similar development process that involved data collection from human annotators. EmotionDict [16] used the Polarity Clues [41] as a resource to build upon and was semi-automatically enriched with synonyms. Consequently, the words in these two dictionaries are expected to have largely identical sentiment labels. Since a constant was used to impute numerical sentiment values for the sentiment labels, this similarity persists in the cluster analysis.

Morph was created in an attempt to model the polarity of low-frequency complex German compound words based on their morphological composition. Its base data set is sampled out of the Polart lexicon [15]. In addition, the authors added words from the CELEX database and used additional compound words from Wegwarte, an online collection of German neologisms and Wiktionary. All these words are included in our analysis, as well as the *test-train* set and *dev* set that were used by Ruppenhofer, Steiner and Wiegand [33] to evaluate their approach.

ALPIN (Austrian Language Polarity in Newspapers) [17] was developed in the framework of the DYSEN project[2]. It is the only dictionary that is specific to Austria. The labeled text data that was used for creating it stems from Austrian newspapers and contains, *inter alia*, German words specific to Austria. Its relation to Austria sets it apart, and it is was developed independently of already existing resources. Unsurprisingly, it was separated rather early in the clustering process, indicating that is rather different from the other dictionaries within the large cluster on the left-hand side of the dendrogram in Figure 2.

SentiMerge [11] was created based on a Bayesian probabilistic model and combines polarity values from the Polart lexicon [15], SentiWS [29], Polarity Clues [41] and the German SentiSpin dictionary [42]. The latter resource was not included in the analysis at hand as it was not accessible and the author was not reachable. In the cluster dendrogram, SentiMerge appears in close proximity to two of its constituents, SentiWS [29] and Polarity Clues [41]. Interestingly, the Polart lexicon is very distant from all other German polarity resources and forms a cluster of its own (see below).

The Sentiment Phrase List (SePL) [31] was the second dictionary in the clustering process to initiate a splinter group and is thus quite dissimilar from the remaining dictionaries. This does not come as a surprise, as it possesses some unique features that set it apart. First, it was created based on product reviews accompanied with one to five-star ratings. Second, it contains not only single words, but short phrases like *absoluter Mist* ("absolute rubbish"). It can thus be expected to have a small overlap with the other dictionaries.

The Polart lexicon forms its own cluster and was the first object to be separated into a splinter group during the iterative clustering process. This is surprising, as some dictionaries, like Morph, used the Polart lexicon as a seed dictionary. The dissimilarity to the other dictionaries might be attributed to the interesting structure of Polart. It provides categorical labels that indicate sentiment orientation as well as numerical values that indicate sentiment

[2] *Dynamic Sentiment Analysis as Emotional Compass for the Digital Media Landscape*; more information on the DYSEN project can be found here: `https://www.oeaw.ac.at/acdh/projects/dysen/`.

intensity. The sentiment intensity, however, can take seven different, discrete values: 0, 0.3, 0.5 and 0.7, as well as their negatives. This sets it apart from the other dictionaries: It is less fine-grained than the dictionaries that contain continuous sentiment values, but more fine-grained than the dictionaries that only provide sentiment orientation in two or three categories and that had to be imputed with the positive and negative mean sentiment value calculated from the numerical sentiment dictionaries. Thus, Polart being an outgroup in the dendrogram may be a reflex of this scaling in combination with the use of (distributionally sensitive) Euclidean distance for clustering dictionaries.

5 Conclusion and Outlook

In this paper, we present an overview of a lion's share of the German sentiment dictionaries that are currently available. It becomes evident that polarity resources are very heterogeneous both in terms of how they are generated and their structure. Although it is reassuring to see that most of them share similarities as to how words are rated, we also see that some of them form subgroups consisting of dictionaries that depend on each other.

These dependencies are an immediate consequence of the compilation procedure. First, some dictionaries are direct extensions of others. Second, extant dictionaries are often used to evaluate new dictionaries. This can be potentially problematic: if new dictionaries are only tested against extant resources that are already related, this may in the worst case amplify built-in biases and propagate labeling errors. We thus recommend using diverse polarity resources both for the evaluation of new sentiment dictionaries as well as, more generally, for testing and evaluating sentiment-analysis algorithms.

The "dictionary of dictionaries" we assembled during the research process is publicly available for further research and can be accessed on the Gitlab repository for this paper[3] or on Github[4].

This resource is not meant as a ready-to-use tool for sentiment analysis. It is rather a by-product of our research process and made available to encourage and facilitate further research on German polarity resources for sentiment analysis. Numerous compelling research question might be investigated with the help of our dictionary of dictionaries that the scope of our paper did not touch upon.

For one, we did not compare the performance of individual or subgroups of resources with each other. Recent research has shown that side-by-side performance comparisons of off-the-shelf sentiment resources can give fruitful insights into their reliability and validity [5].

Secondly, we focused on polarity ratings and discarded other sentiment dimensions if they were available. This was done to facilitate comparisons of the highly heterogeneous resources. It may be worthwhile for future research to evaluate the benefits of a multi-dimensional approach in sentiment analysis. Thirdly, our aim was not to create an integrated dictionary, but to bring the available dictionaries into a comparable format. The authors of SentiMerge [11] propose a bayesian framework for dictionary integration to deal with differences between individual dictionaries via statistical modeling. The dictionary assembled by us opens up a convenient and comprehensive framework to test and apply such and similar approaches in future research.

And finally, while we note that the dependencies among German polarity resources may be problematic with regard to bias propagation, we do not evaluate or quantify potential

[3] https://gitlab.com/acdh-oeaw/dysen/dysen-ldk2021
[4] https://github.com/bettina-mj-kern/LDK_2021

bias in any way, as this is beyond the scope of this paper. The detection of bias is, however, undoubtedly an important issue in sentiment analysis and requires further research with regard to how to identify and remedy biases in sentiment tools. Recent research indicates that the validity of sentiment resources is in many cases questionable [39]. Moreover, there are reasonable doubts about whether sentiment dictionaries should be applied outside the domain or even the intended use case for which they were developed [30]. During the initial emergence of sentiment analysis, the development focus was primarily on the scalability of the tools, on their ability to harness large amounts of text in an automated fashion and draw information from them. As the field advances and matures, validity, reliability and risk of bias emerge as relevant areas of research with the aim to attain more robust and more fine-grained results that accurately and reliably capture the sentiment content in a text. Our study provides only a piece of the mosaic and hopefully gives rise to further research.

References

1 Sattam Almatarneh and Pablo Gamallo. Automatic construction of domain-specific sentiment lexicons for polarity classification. In *Trends in Cyber-Physical Multi-Agent Systems. The PAAMS Collection - 15th International Conference, PAAMS 2017*, pages 175–182. Springer International Publishing, 2018. doi:10.1007/978-3-319-61578-3_17.

2 Jens Ambrasat, Christian von Scheve, Markus Conrad, Gesche Schauenburg, and Tobias Schröder. Consensus and stratification in the affective meaning of human sociality. *Proceedings of the National Academy of Sciences*, 111(22):8001–8006, 2014.

3 Olatz Arbelaitz, Ibai Gurrutxaga, Javier Muguerza, Jesús M Pérez, and Inigo Perona. An extensive comparative study of cluster validity indices. *Pattern Recognition*, 46(1):243–256, 2013. doi:10.1016/j.patcog.2012.07.021.

4 Guy Brock, Vasyl Pihur, Susmita Datta, and Somnath Datta. clValid: An R package for cluster validation. *Journal of Statistical Software*, 25(4):1–22, 2008. URL: http://www.jstatsoft.org/v25/i04/.

5 Chung-hong Chan, Joseph Bajjalieh, Loretta Auvil, Hartmut Wessler, Scott Althaus, Kasper Welbers, Wouter van Atteveldt, and Marc Jungblut. Four best practices for measuring news sentiment using 'off-the-shelf'dictionaries: a large-scale p-hacking experiment. *Computational Communication Research*, 3(1):1–27, 2021. doi:10.5117/CCR2021.1.001.CHAN.

6 Mark Cieliebak, Jan Milan Deriu, Dominic Egger, and Fatih Uzdilli. A twitter corpus and benchmark resources for german sentiment analysis. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 45–51, 2017.

7 Evgenia Dimitriadou, Sara Dolničar, and Andreas Weingessel. An examination of indexes for determining the number of clusters in binary data sets. *Psychometrika*, 67(1):137–159, 2002.

8 Elizabeth Duffy. Emotion: an example of the need for reorientation in psychology. *Psychological Review*, 41(2):184, 1934. doi:10.1037/h0074603.

9 Joseph C Dunn. Well-separated clusters and optimal fuzzy partitions. *Journal of cybernetics*, 4(1):95–104, 1974. doi:10.1080/01969727408546059.

10 Paul Ekman. Basic emotions. *Handbook of cognition and emotion*, 98(45-60):16, 1999.

11 Guy Emerson and Thierry Declerck. Sentimerge: Combining sentiment lexicons in a bayesian framework. In *Proceedings of Workshop on Lexical and Grammatical Resources for Language Processing*, pages 30–38, 2014.

12 Julia Handl, Joshua Knowles, and Douglas B Kell. Computational cluster validation in post-genomic data analysis. *Bioinformatics*, 21(15):3201–3212, 2005. doi:10.1093/bioinformatics/bti517.

13 Philipp Kanske and Sonja A Kotz. Leipzig affective norms for german: A reliability study. *Behavior research methods*, 42(4):987–991, 2010. doi:10.3758/BRM.42.4.987.

14 Leonard Kaufman and Peter J Rousseeuw. *Finding groups in data: an introduction to cluster analysis*, volume 344. John Wiley & Sons, 2009.

15 Manfred Klenner, Angela Fahrni, and Stefanos Petrakis. Polart: A robust tool for sentiment analysis. In *Proceedings of the 17th Nordic Conference of Computational Linguistics (NODALIDA 2009)*, pages 235–238, 2009.

16 Roman Klinger, Surayya Samat Suliya, and Nils Reiter. Automatic emotion detection for quantitative literary studies. a case study based on franz kafka's "das schloss" and "amerika". *Proceedings of the Digital Humanities*, 2016.

17 Thomas Kolb, Katharina Sekanina, Andreas Baumann, and Julia Neidhardt. Austrian language polarity in newspapers (ALPIN). Dataset, v1.0. URL: `https://phaidra.univie.ac.at/o:1169855`.

18 Maximilian Köper and Sabine Schulte Im Walde. Automatically generated affective norms of abstractness, arousal, imageability and valence for 350 000 german lemmas. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2595–2598, 2016.

19 Olaf Lahl, Anja S Göritz, Reinhard Pietrowsky, and Jessica Rosenberg. Using the world-wide web to obtain large-scale word norms: 190,212 ratings on a set of 2,654 german nouns. *Behavior Research Methods*, 41(1):13–19, 2009. `doi:10.3758/BRM.41.1.13`.

20 Peter Lang. Behavioral treatment and bio-behavioral assessment: Computer applications. *Technology in mental health care delivery systems*, pages 119–137, 1980.

21 Martin Maechler, Peter Rousseeuw, Anja Struyf, Mia Hubert, and Kurt Hornik. *cluster: Cluster Analysis Basics and Extensions*, 2019.

22 Ujjwal Maulik and Sanghamitra Bandyopadhyay. Performance evaluation of some clustering algorithms and validity indices. *IEEE Transactions on pattern analysis and machine intelligence*, 24(12):1650–1654, 2002. `doi:10.1109/TPAMI.2002.1114856`.

23 Glenn W Milligan and Martha C Cooper. An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50(2):159–179, 1985. `doi:10.1007/BF02294245`.

24 Charles E Osgood. Dimensionality of the semantic space for communication via facial expressions. *Scandinavian journal of psychology*, 7(1):1–30, 1966. `doi:10.1111/j.1467-9450.1966.tb01334.x`.

25 Charles E Osgood, George J Suci, and Percy H Tannenbaum. 1957the measurement of meaning. *Urbana: University of Illinois Press*, 47, 1957.

26 Vasyl Pihur, Somnath Datta, and Susmita Datta. *RankAggreg: Weighted Rank Aggregation*, 2020. R package version 0.6.6. URL: `https://CRAN.R-project.org/package=RankAggreg`.

27 R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2020. URL: `https://www.R-project.org/`.

28 Jutta Ransmayr, Karlheinz Mörth, and Matej Ďurčo. Ii. amc (austrian media corpus) – korpusbasierte forschungen zum österreichischen deutsch, 2017.

29 Robert Remus, Uwe Quasthoff, and Gerhard Heyer. Sentiws – a publicly available german-language resource for sentiment analysis. In *Proceedings of the 7th International Language Resources and Evaluation (LREC'10)*, pages 1168–1171, 2010.

30 Filipe N Ribeiro, Matheus Araújo, Pollyanna Gonçalves, Marcos André Gonçalves, and Fabrício Benevenuto. Sentibench-a benchmark comparison of state-of-the-practice sentiment analysis methods. *EPJ Data Science*, 5(1):1–29, 2016. `doi:10.1140/epjds/s13688-016-0085-1`.

31 Sven Rill, Sven Adolph, Johannes Drescher, Dirk Reinel, Jörg Scheidt, Oliver Schütz, Florian Wogenstein, Roberto V Zicari, and Nikolaos Korfiatis. A phrase-based opinion list for the german language. In *Proceedings of the 11th Conference on Natural Language Processing (KONVENS'2012)*, pages 305–313, 2012.

32 Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987. `doi:10.1016/0377-0427(87)90125-7`.

33 Josef Ruppenhofer, Petra Steiner, and Michael Wiegand. Evaluating the morphological compositionality of polarity. In *Proceedings of the 11th international conference on Recent Advances in Natural Language Processing (RANLP'2017)*, pages 625—633, 2017.

34 James A Russell. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161, 1980.

35 David S Schmidtke, Tobias Schröder, Arthur M Jacobs, and Markus Conrad. Angst: Affective norms for german sentiment terms, derived from the affective norms for english words. *Behavior research methods*, 46(4):1108–1118, 2014. doi:10.3758/s13428-013-0426-y.

36 Tobias Schröder. A model of language-based impression formation and attribution among germans. *Journal of Language and Social Psychology*, 30(1):82–102, 2011. doi:10.1177/0261927X10387103.

37 Peter Turney, Yair Neuman, Dan Assaf, and Yohai Cohen. Literal and metaphorical sense identification through concrete and abstract context. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 680–690, 2011.

38 Peter D Turney and Michael L Littman. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems (TOIS)*, 21(4):315–346, 2003. doi:10.1145/944012.944013.

39 Wouter van Atteveldt, Mariken ACG van der Velden, and Mark Boukes. The validity of sentiment analysis: Comparing manual annotation, crowd-coding, dictionary approaches, and machine learning algorithms. *Communication Methods and Measures*, pages 1–20, 2021. doi:10.1080/19312458.2020.1869198.

40 Melissa LH Vo, Markus Conrad, Lars Kuchinke, Karolina Urton, Markus J Hofmann, and Arthur M Jacobs. The berlin affective word list reloaded (bawl-r). *Behavior research methods*, 41(2):534–538, 2009. doi:10.3758/BRM.41.2.534.

41 Ulli Waltinger. German polarity clues: A lexical resource for german sentiment analysis. In *LREC*, pages 1638–1642. Citeseer, 2010.

42 Ulli Waltinger. Sentiment analysis reloaded-a comparative study on sentiment polarity identification combining machine learning and subjectivity features. In *WEBIST (1)*, pages 203–210, 2010.

43 Hadley Wickham. *stringr: Simple, Consistent Wrappers for Common String Operations*, 2019. R package version 1.4.0. URL: https://CRAN.R-project.org/package=stringr.

A Preprocessing of the Sentiment Dictionaries

AffDict [36]. This dictionary contains word ratings three dimensions on social concepts, separately for men and women and summarised for both genders. For our purposes, only the word column and evaluation column with averaged ratings by both genders were identified as relevant. The other columns were dropped. The values are scaled [−4,4] and thus scaled to [−1,1]. The German umlauts (ae → ä, oe → ö and ue → ü) were changed manually inside the csv file, as there are words that contain these letter combinations, but are not umlauts (e.g. homosexuell) which makes it difficult to find a regular expression pattern. Placeholders like "versprechen *(etwas)*", "zanken *mit*", "lernen *von*" were removed with regular expressions to match the other dictionaries.

AffMeaning [2]. The AffMeaning was created while investigating the affective meaning of authority- / community-related concepts. 2849 participants rated 909 words on three dimensions. The ratings are averaged separately for men and women and for both genders. For our purposes, only the evaluation columns with ratings by both genders were identified as relevant, other columns were dropped. A 9-point semantic differential scale was used and averaged over participants. The values thus range from [1, 8] and were scaled to [−1,1] for

the purpose of our analysis. Placeholders like *jmd. auszeichnen* were removed with regular expressions.

AffNorms [18]. The resource represents the most extensive sentiment dictionary for German at the current time. It consists of 350 000 German lemmas with four psycho-linguistic attributes: abstractness, arousal, imageability and valence. For the purpose of this paper, only the valence scale was used. It reflects polarity scaled [0,10] and was rescaled to [−1,1].

ALPIN [17]. This dictionary is based on roughly 5000 labeled text snippets taken from the Austrian Media Corpus [28] that were labeled in a crowd-sourcing survey. Sentiment scores range in the interval [−1, 1]. Only words that surface in more than one snippet were included resulting in 4600 words in total before further cleaning. During preprocessing, abbreviations, urls, numbers and symbols were removed. For duplicate words with different sentiment score, the mean was taken.

ANGST [35]. The aim of the Affective Norms for German Sentiment Terms (ANGST) was to provide a German adaptation of the ANEW, the Affective Norms for English Words [20]). This corpus provides normative emotional ratings of pleasure, arousal and dominance for a large number of English words. BAWL-R was used as a starting groundwork. In the case of the valence, the ANEW words were translated into German and received ratings from BAWL-R if available. For words not in the BAWL-R, ratings on a bipolar scale ranging $[-3, 3]$ were collected from 65 participants. For our study, the word column and the valence column were extracted from the data set. The ratings were scaled to [−1,1]. Further preprocessing was not necessary.

BAWL-R [40]. The aim of BAWL-R is to help create stimulus material for experiments on affective verbal processing. It is a revision and extension of the previous version, BAWL: 700 new words and arousal ratings were added by surveying 200 Psychology students. BAWL-R contains ratings for imageability, arousal and valence for 2900 words, as well as standard deviations and meta data such as the number of letters, syllables and phonemes, bigram frequencies, number of orthographic neighbours and so on. For our purposes, the word column and the corresponding valence values were extracted. Nouns were capitalised using the `str_to_title()` function from the `stringr` package [43]. The valence ratings are ranged [−3,3] and were thus rescaled to [−1,1].

EmotionDict. Extension of sentiment analysis to literary texts. EmotionDict follows Ekman's definition of fundamental emotions [10]. This is of note because most other sentiment dictionaries follow a dimensional approach. Ekman's theory, in contrast, describes emotions as discrete categories, distinguishable by an individual's facial expression and physiological patterns, for example of the autonomic nervous system. EmotionDict consists of seven text files, containing words that reflect the seven basic emotions: anger (*Wut*), fear (*Angst*), enjoyment (*Freude*), sadness (*Trauer*), disgust (*Ekel*), contempt (*Verachtung*) and surprise (*Überraschung*). The surprise text file had to be excluded as it contains a mixture of words with different polarities. The remaining six text files were merged into a single word list. Words in the joy text file were assigned a positive numerical value and words in the other five text files received the negative numerical value, as described in the main text.

LANG. The Leipzig Affective Norms for German [13] was created to provide researchers with norms for experimental studies on verbal emotional processing. 1 000 short German nouns were rated twice by two independent samples two years apart to assess the retest reliability

across samples and time. The rating was done on a 9-point scale using self-assessment manikins. Only the ratings on valence were used for this paper. Originally ranging from 1 to 9, the ratings were rescaled to values between −1 and 1.

Morph [33]. This dictionary was created researching coverage problems of German sentiment dictionaries.The authors attempt to estimate the polarity of complex German compound words based on the polarity of their morphological composition. This resource was not meant as a tool for sentiment analysis, but merely presents the result of modelling a word's sentiment based on its constituents. It is therefore questionable how well Morph will perform as a sentiment prediction tool. The dictionary consists of five text files containing words and three sentiment labels (NEG, NEU and POS). One of them contains only affixes and was not included in the analysis. Additional information like word type or inflections were removed with regular expressions. Words with a negative sentiment label were assigned a numerical value as described in the text, and words with a neutral one 0.

PolArt [15]. The Polart lexicon contains negative, neutral and positive sentiment labels and fixed numerical values (0, 0.3, 0.5 and 0.7) that encode sentiment intensity. For negative labels, the sentiment value was reversed to negative by multiplying it with −1. The resource further includes shift words that reverse the polarity if neighbouring words, and intensifiers. For these words, we assigned a neutral sentiment value of 0. The label column was then dropped along with other unneeded columns.

Polarity Clues [41]. The Polarity Clues consist of three text files with negative, positive and neutral lemmas. Three other text files contain their inflected forms, but for our purpose, lemmas are sufficient. The word column and the sentiment labels were extracted for the analysis. The labels were then transformed into numerical values as described in the main text. The first 19 rows were dropped as they contained numbers and symbols that were not relevant for us.

SentiMerge [11]. The authors propose a framework for merging sentiment resources of different lengths and different scales. The authors demonstrate their method by merging the Polart lexicon, SentiWS, Polarity Clues and the German SentiSpin [42] that was also used to build the Polarity Clues. The words in the dictionary were all lowercase and thus transformed to title case in all instances that that had the "noun" part-of-speech tag, again making use of Hadley Wickham's `stringr` package [43]. 878 words, however, were tagged as "XY" and remained lowercase. 1805 words appeared between two and four times in the dictionary because they exhibited different part-of-speech tags which probably originated from the merging of different sentiment resources. We resolved this issue by taking the mean sentiment of these words and discarding the superfluous entries.The values were rescaled to [−1, 1].

SentiWS [29]. SentiWS contains 3 471 negative and positive words, their inflections, part-of-speech tags and a sentiment value between −1 and 1. The negative and positive words come in two different text files. After some light data cleaning and removal of unneeded columns, the two sets are combined into one. Further preprocessing was not necessary.

SePL [31]. The Sentiment Phrase List provides opinion values ranging [−1, 1] for words and short phrases, as well as standard deviations and standard errors. The relevant column

containing the opinion value was extracted from the data in the text file alongside with the corresponding words. Further preprocessing steps were not necessary.

Wordnorms [19]. The Wordnorms consist of 2654 nouns that were rated on concreteness, valence and arousal by a sizable sample of 3 907 participants via web application. The resulting sentiment dictionary contains standard deviations for the mean ratings as well as metadata, like the number of ratings each word received, the number of letters and the results of the cluster analysis. For our research interest, only the words and mean valence ratings were relevant. Words without a valence rating were dropped. The ratings ranged $[0, 5]$ and were consequently scaled to $[-1,1]$.

Exploring Causal Relationships Among Emotional and Topical Trajectories in Political Text Data

Andreas Baumann
University of Vienna, Austria

Klaus Hofmann
University of Vienna, Austria

Bettina Kern
University of Vienna, Austria

Anna Marakasova
TU Wien, Austria

Julia Neidhardt
TU Wien, Austria

Tanja Wissik
Austrian Academy of Sciences, Vienna, Austria

Abstract

We explore relationships between dynamics of emotion (arousal and valence) and topical stability in political discourse in two diachronic corpora of Austrian German. In doing so, we assess interactions among emotional and topical dynamics related to political parties as well as interactions between two different domains of discourse: debates in the parliament and journalistic media. Methodologically, we employ unsupervised techniques, time-series clustering and Granger-causal modeling to detect potential interactions. We find that emotional and topical dynamics in the media are only rarely a reflex of dynamics in parliamentary discourse.

2012 ACM Subject Classification Computing methodologies → Lexical semantics; Computing methodologies → Discourse, dialogue and pragmatics; Information systems → Sentiment analysis

Keywords and phrases time-series clustering, Granger causality, topical stability, emotion, political discourse

Digital Object Identifier 10.4230/OASIcs.LDK.2021.38

Funding This research was funded by the Austrian Academy of Sciences (go!digital Next Generation grant, GDNG 2018-020) and by the City of Vienna (Digitaler Humanismus grant, MA7-737909/19).

1 Introduction

Political discourse is evidently associated with emotions [10]. As new topics emerge they are, for example, framed positively or negatively by political stakeholders in their communication, and these dynamics are received and perhaps even amplified by the media [18, 22]. In this paper, we explore to what extent shifts in the topics that political parties are associated with drive or are in fact driven by emotional dynamics. We do so in an explicitly exploratory way; after all, it is hard to evaluate causal relationships between topical and emotional dynamics. More concretely, we analyze time series that characterize dynamics of (i) emotional valence (ii) arousal and, (iii) topical stability, for three political parties in the Austrian parliament.

To tackle interactions among discourse in the parliament and in the media we investigate two corpora as part of our ongoing project DYLEN [1]: the ParlAT corpus of parliamentary speeches in Austria and the Austrian Media Corpus, covering both print and online media. Since we are interested in the dynamic aspects of the interaction between topical stability

and emotion, we adopt a diachronic approach, covering a period of 20 years. In our analysis, we first identify which of the variables display similar diachronic dynamics and subsequently map interactions among the variables in networks based on Granger causality [4, 28]. This allows us to assess, for example, whether parliamentary debates drive discourse in the media (or vice versa), and whether the emotions encoded in the language associated with one party are significantly driven by the dynamics in the language of another party. For example, it is not *a priori* clear whether emotions in the discourse by right-wing parties are a consequence of topical shifts by left-wing parties, or conversely, whether the former in fact drives topical changes in political discourse. More generally, it is interesting to investigate whether long-term dynamics in the media are at all related to dynamics in parliamentary discourse, or whether the two domains may better be described as dissociated spheres of political discourse. We argue that this data-driven exploratory approach has the potential of generating interesting hypotheses that can (and should) subsequently be evaluated in more detailed (qualitative) investigations.

The application of Granger causality to investigate the impact of sentiment (or more generally: emotion) in texts on variables of interest is not new. In particular, Granger causality was used to predict trends in economics and finance based on sentiment encoded in tweets [17, 11] or in newspaper articles [9]. In research on health-care, Granger-causal modeling revealed interactions among sentiment and the tendency to participate in medically related discussions on Reddit [2], as well as effects of anxiety dynamics in tweets on changes in social-interaction behavior [3]. On the structural level of language, Granger casuality was employed to analyze the relationship between syntactic change and frequency [16]. In our contribution, we focus on the interaction between emotion and topical shifts in the political context.

We first describe our data and how diachronic trajectories for topical stability and emotion estimates were derived. We then present the pipeline of our exploratory analysis resulting in Granger-causal networks. Finally, we briefly discuss our results as well as possible future directions of our research.

2 Data and time-series pre-processing

The data analyzed in our study comes from two different sources: first, the Austrian Corpus of Parliamentary Records (ParlAT; [25]), consisting of transcribed speeches in the Austrian parliament; second, the Austrian Media Corpus (AMC; [15]) consisting of Austrian print and online media. For the present analysis, both corpora were limited to the period from 1997 to 2016, thus covering two decades of political discourse in Austria. The two corpora differ considerably in size and structure. While ParlAT consists of 75 million tokens, AMC is much bigger covering 5.5 billion tokens. Even though in their current form the corpora do not allow us to track causal dynamics within the time frame of individual news cycles (e.g. interviews and opinion pieces by influential figures and "spin doctors" are not tagged separately in AMC), the two corpora combined provide the best available coverage of Austrian political discourse to explore questions on a broader temporal scale.

Both corpora were used to derive time series for three different variables and three different groups of individuals, namely the political parties FP ("Freiheitliche Partei Österreichs", Austrian freedom party; right-wing), VP ("Österreichische Volkspartei", Austrian people's party; conservatives), and SP ("Sozialdemokratische Partei Österreichs", Austrian social democrats). The three variables considered are (i) topical stability, (ii) valence, and (iii) arousal. Valence and arousal refer to two emotional dimensions; while valence measures whether a text is

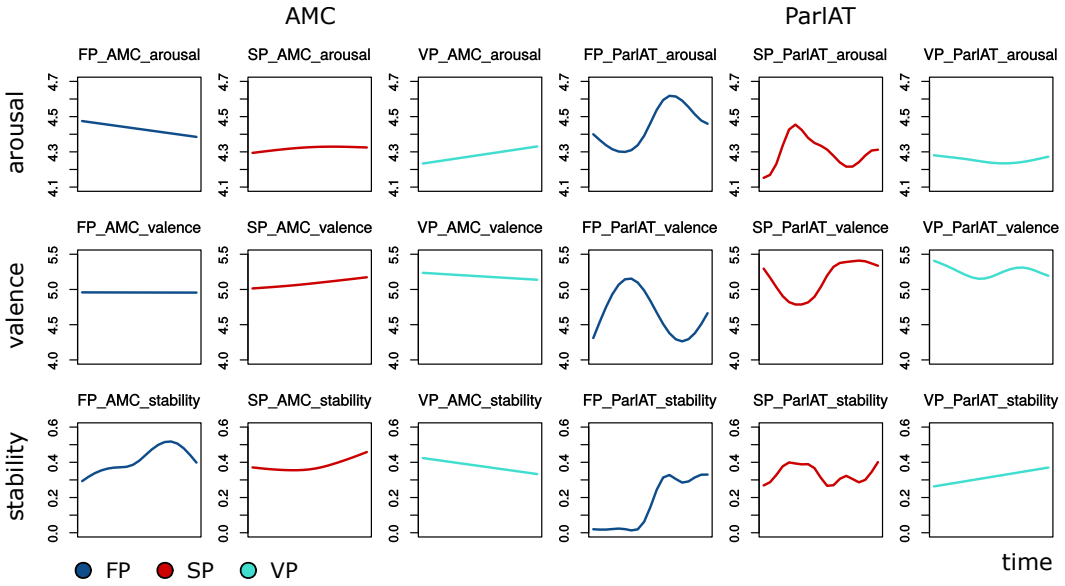


Figure 1 Trajectories for all corpora, metrics and targets (SP: red; FP: blue; VP: turquoise).

negative or positive, arousal measures the extent to which the text represents calm or agitated language [24, 21]. For valence and arousal, we used the time series computed in [7]. In a nutshell, these time series were determined by splitting both corpora into sub-corpora for each year, then, in each sub-corpus, extracting 200 words that are distinctive for each party (FP, VP, SP). Distinctive words were determined based on chi-squared statistics (see [5] for details). After that, a sentiment dictionary [8] was used to determine average valence and arousal, respectively, based on the sets of distinctive lexical items, in an unsupervised fashion. This was done separately for every single year.

To illustrate this approach, let us consider the three most positive and most negative words, respectively, associated with FP in ParlAT in two different years. In 2004, the most negative words are *Kriminalität* ("crime", valence: 1.77), *Vorwurf* ("allegation", 1.78), and *Opfer* ("victim", 2.48), while the most positive words are *Freude* ("joy", 8.29), *Erfolg* ("success", 8.23), and *Gute* ("the good (one)", 7.67). In contrast, the most negative words associated with FP in 2010 are *Versagen* ("failure", 1.17), *Verfassungsbruch* ("constitutional violation", 1.52), and *Arbeitsverweigerung* ("refusal to work", 1.76); the most positive ones are *Familie* ("family", 7.55), *Mut* ("courage", 7.38), and *Wein* ("wine", 7.04). Crucially, the most extreme words in 2004 show a higher valence than those in 2010. Differences at the 10^{-1} level already indicate a noticeable shift in this regard. In the time series of year-wise average valence/arousal scores over all 200 distinctive words, such shifts are encoded for the whole observation period.

In total, 12 time series were generated as described above (two corpora, three parties, two emotion scores). To eliminate noise, generalized additive models (GAM; [26]) were fitted to each time series. GAMs are suitable models for analyzing the time series at hand since they also capture non-linear dynamics (and, moreover, allow for factoring in autocorrelation). The predicted values of the fitted GAMs were then used for the present analysis. More details on the modeling procedure can be found in [7].

We measured topical stability of the parties over time by applying Jaccard index [6] to compare semantic neighbourhoods of the party names in two subsequent years. First, we computed word co-occurrence matrices with positive pointwise mutual information (PPMI)

scores of each yearly subcorpus of the two corpora (AMC and ParlAT). The subcorpora are lemmatised, and only nouns, verbs and adjectives were considered. PPMI vector representation was preferred over state-of-the-art dense embeddings (i.e. word2vec or GloVe vectors) due to the fact that the latter resulted in largely linear dynamics (see below). Thus, for each party name in each year we extract semantic neighborhood which is represented by top-n semantically most similar words. The size of the neighbourhood is set to 50 words.[1] Next, each neighbourhood set is compared to the one from the previous year using Jaccard index which, in total, results in six additional time series.

Again, GAMs were fitted to each of the six time series in order to take care of noisy data, and the GAM estimates were added to our data set for further analysis. Thus, our final dataset consists of time series for 18 variables, each made up of scores for 20 years. The 18 resulting time series are shown in Figure 1. The dataset can be downloaded from `https://phaidra.univie.ac.at/o:1168825`. It can be seen, for example, that valence in FP discourse in the parliament seems to peak around 2004 and subsequently drops to obtain its minimum in 2010 (in [7] we argue that this might be a consequence of the transition of FP from government to opposition; an extra-linguistic factor which is certainly relevant, as one of the reviewers has pointed out as well).

3 Analysis

Our analysis unfolds in three steps: first, clustering of the time series described in the previous section; second, identification of clusters; third, computation of Granger-causal networks based on the clusters. Each of these steps will be explained in more detail in the following.

In order to cluster the time-series of emotional scores and topical stability, we first need a distance measure to assess the degree to which two time series are similar to each other. We opted for autocorrelation function (ACF) distance, which is derived by first computing the ACF for each time series and then, for each pair, computing the Euclidean distance between the two time series [12]. Thus, ACF distance treats those time series as similar which have a similar autocorrelation structure, i.e. which are characterized by similar degrees of changeability through time. This measure has multiple advantages for analyzing our data. First, it implicitly normalizes all variables, which is important since arousal, valence and topical stability operate on different scales. Second, it is invariant with respect to the orientation of the observed variable. That is, if a time series has, say, a W-shaped curve, then this time series and its vertically flipped M-shaped variant have the same ACF and are hence treated as similar. This is important for our analysis, since we also want to detect if downward trends in one variable are linked to upward trends in another variable. Since linear time series have identical and linearly decreasing ACFs, only non-linear time series were included in the analysis. This will be important for the causal analysis explained below, since the causal methods employed in this paper do not reasonably apply to pairs of linear time series.

We used ACF distance to derive a distance matrix for the remaining 12 variables (i.e. time series) in our dataset. We then applied hierarchical agglomerative clustering to this distance matrix to identify groups of similarly behaving time series. We used Ward linkage as a clustering criterion [27] and determined the optimal number of clusters through maximizing

[1] Experiments with the larger neighbourhood sizes did not show significant difference with respect to the current findings.

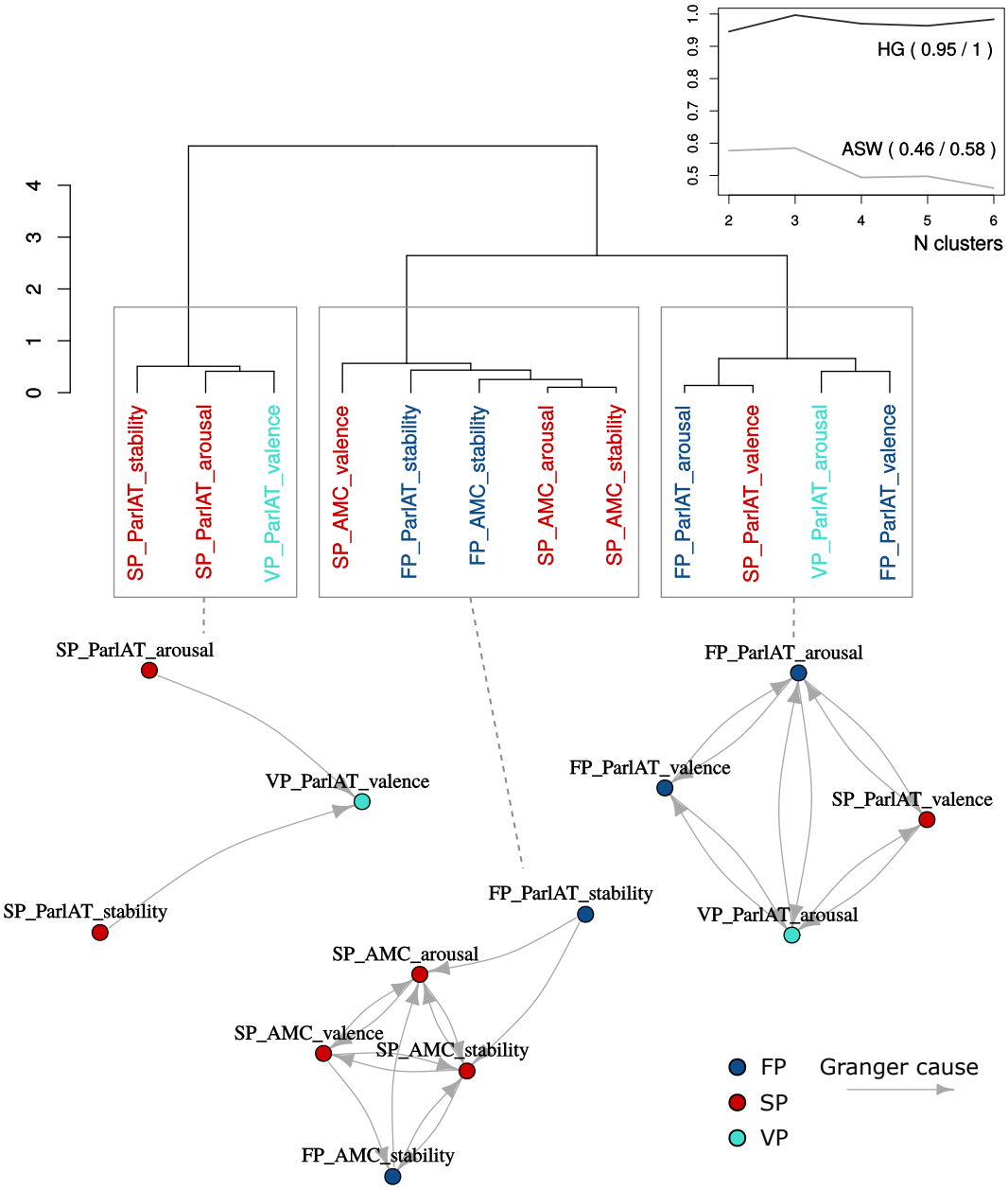


Figure 2 Top left: Hierarchical clustering of all time series (driven by ACF distance; complete linkage). Top right: Clustering quality measures average silhouette width (ASW) and Hubert's Gamma (HG). Bottom: Granger-causal networks for the clusters in the dendrogram. Each node represents a time series. Two nodes are linked if one node significantly Granger-causes another node. Arrows denote causal relationships pointing from cause to effect. Color code: SP: red; FP: blue; VP: turquoise.

average silhouette width (ASW), measuring homogeneity of clusters, and Hubert's Gamma (HG), which measures the extent to which the dendrogram reflects the original distance matrix [19]. A robustness analysis involving other linkage methods (single, average, complete, median, centroid) revealed that the final clusters are in fact invariant with respect to linkage selection. The optimal number of clusters in the dendrogram was computed as three. The resulting dendrogram and the corresponding measures for clustering quality are shown in Figure 2 (top).

After that, causal networks were computed for each cluster separately by means of Granger causality tests [4, 28]. Granger causality is a concept for modelling the causal relationship between two time series x_t and y_t. The underlying idea is to check whether predicting y_t is significantly improved by also considering past information of the former time series, i.e. x_{t-k} for some lag k (based on a Wald test). If this is the case then x_t is said to Granger cause y_t (but not necessarily vice versa). In the present analysis, we opted to only consider past information that goes back up to one time step (lag $k = 1$), i.e. one year, since we do not consider interactions among instances of discourse over more than one year plausible. More fine-grained time scales might plausibly allow for more than one time step. In each cluster in the dendrogram, a Granger test was computed for each pair of time series in that cluster, thereby considering both potential directions of causality. A significance threshold of $\alpha = 0.05$ was employed, which was Bonferroni-corrected through the overall number of Granger tests in that cluster (note, however, that employing a fixed threshold of $\alpha = 0.01$ yields exactly the same qualitative results; cf. e.g. [14]). Subsequently, a directed Granger causal graph was created for each cluster, in which two variables are linked (from cause to effect) if the p-value of their corresponding Granger test is below the respective significance threshold.

Figure 2 (bottom) shows Granger-causal networks for the three clusters. The left-most cluster displays causal relationships from SP stability and SP arousal to VP valence (all in ParlAT). The cluster in the middle shows all SP variables and FP stability in AMC being mutually connected and Granger-causally affected by FP stability in ParlAT. The final cluster shows mutually connected emotion variables (valence, arousal) for all parties in ParlAT.

4 Discussion and outlook

In this paper, we have shown how interactions among dynamics of emotion encoded in political discourse and topical changes, both across political parties and domains (parliamentary speeches; media) can be analyzed by means of time-series analysis and Granger-causal modeling, thus extending the application of Granger causality to the analysis of political text data.

Two observations can be made: First, it can be seen that topical stability and emotion are interconnected. However, we do not see a clear tendency that emotional shifts are driven by topical changes or vice versa. Second, the two domains, parliamentary discourse and media, seem to be rather disconnected. The only exception to this rule is topical stability of FP in the parliament which seems to affect dynamics in the media (both topical and emotional). This is interesting and tentatively suggests that changes in contributions to parliamentary discourse by right-wing politicians functions as an important driver of dynamics in the media. Conversely, it might be the case that right-wing discourse is more likely to be picked up and reflected on by Austrian media outlets (also if reporting on left-wing discourse) than the discourse produced by other parts of the political spectrum [23].

However, this observation has to be treated with caution. Evidently, our approach has many shortcomings. First, our diachronic data is rather coarsely grained. For analyzing time series in linguistic dynamics both a longer time span and shorter subperiods (e.g. months instead of years) are desirable to obtain more robust results. Second, the estimation of the variables investigated (valence, arousal, stability) was based on rather simple and straightforward methods, which were motivated by the large structural difference between the two underlying corpora (AMC vs. ParlAT). There is undoubtedly room for more sophisticated and reliable methods for emotion and topical change detection. Third, it is evident that

Granger causality is only one model of what is usually conceptualized as causality. It will be interesting to compare whether other methods for detecting causal relationships among time series, like Bayesian dynamic networks [13] or convergent cross mapping [20], produce similar outcomes. Still, we find that our exploratory approach generates stimulating hypotheses that deserve further investigation in future studies.

References

1 Andreas Baumann, Julia Neidhardt, and Tanja Wissik. DYLEN: Diachronic Dynamics of Lexical Networks. In *LDK (Posters)*, pages 24–28, 2019.

2 Giovanni Delnevo, Marco Roccetti, and Silvia Mirri. Modeling patients' online medical conversations: a granger causality approach. In *Proceedings of the 2018 IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies*, pages 40–44, 2018.

3 Sarmistha Dutta, Jennifer Ma, and Munmun De Choudhury. Measuring the impact of anxiety on online social interactions. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 12, 2018.

4 C. W. J. Granger. Investigating Causal Relations by Econometric Models and Cross-spectral Methods. *Econometrica*, 1969. doi:10.2307/1912791.

5 Klaus Hofmann, Anna Marakasova, Andreas Baumann, Julia Neidhardt, and Tanja Wissik. Comparing lexical usage in political discourse across diachronic corpora. In *Proceedings of the Second ParlaCLARIN Workshop*, pages 58–65, 2020.

6 Paul Jaccard. The distribution of the flora in the alpine zone. 1. *New phytologist*, 11(2):37–50, 1912.

7 Bettina M. J. Kern, Klaus Hofmann, Andreas Baumann, and Tanja Wissik. Komparative Zeitreihenanalyse der lexikalischen Stabilität und Emotion in österreichischen Korpusdaten. In *Proceedings of Digital Lexis and beyond at OELT*, 2021.

8 Maximilian Köper and Sabine Schulte Im Walde. Automatically generated affective norms of abstractness, arousal, imageability and valence for 350 000 german lemmas. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2595–2598, 2016.

9 Jian Li, Zhenjing Xu, Lean Yu, and Ling Tang. Forecasting oil price trends with sentiment of online news articles. *Procedia Computer Science*, 91:1081–1087, 2016.

10 GE Marcus and N Demertzis. *Emotions in politics: The affect dimension in political tension*. Plagrave Macmillan Press, 2013.

11 Anshul Mittal and Arpit Goel. Stock prediction using twitter sentiment analysis. *Standford University, CS229*, 15, 2012. URL: http://cs229.stanford.edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis.pdf.

12 Pablo Montero, José A Vilar, et al. TSclust: An R package for time series clustering. *Journal of Statistical Software*, 62(1):1–43, 2014.

13 Judea Pearl. Graphical models for probabilistic and causal reasoning. In *Computer Science Handbook, Second Edition*. Springer, 2004. doi:10.1201/b16812-50.

14 Thomas V. Perneger. What's wrong with Bonferroni adjustments, 1998. doi:10.1136/bmj.316.7139.1236.

15 Jutta Ransmayr, Karlheinz Mörth, and Matej Ďurčo. Ii. amc (austrian media corpus)–korpusbasierte forschungen zum österreichischen deutsch. In *Digitale Methoden der Korpusforschung in Österreich*. Verlag der Österreichischen Akademie der Wissenschaften, 2017.

16 Malte Rosemeyer and Freek Van de Velde. On cause and correlation in language change: Word order and clefting in brazilian portuguese. *Language Dynamics and Change*, 11(1):130–166, 2021.

17 Jasmina Smailović, Miha Grčar, Nada Lavrač, and Martin Žnidaršič. Predictive sentiment analysis of tweets: A stock market application. In *International Workshop on Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data*, pages 77–88. Springer, 2013.

18 Stefan Stieglitz and Linh Dang-Xuan. Emotions and information diffusion in social media - Sentiment of microblogs and sharing behavior. *Journal of Management Information Systems*, 2013. doi:10.2753/MIS0742-1222290408.

19 Matthias Studer. Weightedcluster library manual: A practical guide to creating typologies of trajectories in the social sciences with r. *LIVES Working papers*, 2013. doi:10.12682/lives.2296-1658.2013.24.

20 George Sugihara, Robert May, Hao Ye, Chih Hao Hsieh, Ethan Deyle, Michael Fogarty, and Stephan Munch. Detecting causality in complex ecosystems. *Science*, 2012. doi:10.1126/science.1227079.

21 Maite Taboada. Sentiment Analysis: An Overview from Linguistics, 2016. doi:10.1146/annurev-linguistics-011415-040518.

22 Joshua Tucker, Andrew Guess, Pablo Barbera, Cristian Vaccari, Alexandra Siegel, Sergey Sanovich, Denis Stukal, and Brendan Nyhan. Social Media, Political Polarization, and Political Disinformation: A Review of the Scientific Literature. *SSRN Electronic Journal*, 2018. doi:10.2139/ssrn.3144139.

23 Ineke Van Der Valk. Right-wing parliamentary discourse on immigration in france. *Discourse & Society*, 14(3):309–348, 2003.

24 Amy Beth Warriner, Victor Kuperman, and Marc Brysbaert. Norms of valence, arousal, and dominance for 13,915 English lemmas. *Behavior Research Methods*, 2013. doi:10.3758/s13428-012-0314-x.

25 Tanja Wissik and Hannes Pirker. ParlAT beta Corpus of Austrian Parliamentary Records. In *Proceedings of the LREC 2018 Workshop'ParlaCLARIN: LREC2018 workshop on creating and using parliamentary corpora*, pages 20–23, 2018.

26 Simon N Wood. *Generalized additive models: an introduction with R*. CRC press, 2017.

27 Mohammed J Zaki and Wagner Meira. *Data mining and analysis: Fundamental concepts and algorithms*. Cambridge University Press, New York, 2014.

28 Cunlu Zou and Jianfeng Feng. Granger causality vs. dynamic bayesian network inference: a comparative study. *BMC bioinformatics*, 10(1):1–17, 2009.

Calculating Argument Diversity in Online Threads

Cedric Waterschoot
KNAW Meertens Instituut, Amsterdam, The Netherlands

Antal van den Bosch
KNAW Meertens Instituut, Amsterdam, The Netherlands

Ernst van den Hemel
KNAW Humanities Cluster NL-Lab, Amsterdam, The Netherlands

Abstract

We propose a method for estimating argument diversity and interactivity in online discussion threads. Using a case study on the subject of Black Pete ("Zwarte Piet") in the Netherlands, the approach for automatic detection of echo chambers is presented. Dynamic thread scoring calculates the status of the discussion on the thread level, while individual messages receive a contribution score reflecting the extent to which the post contributed to the overall interactivity in the thread. We obtain platform-specific results. Gab hosts only echo chambers, while the majority of Reddit threads are balanced in terms of perspectives. Twitter threads cover the whole spectrum of interactivity. While the results based on the case study mirror previous research, this calculation is only the first step towards better understanding and automatic detection of echo effects in online discussions.

2012 ACM Subject Classification Information systems → World Wide Web; Information systems → Sentiment analysis; Human-centered computing → Social media

Keywords and phrases Social Media, Echo Chamber, Interactivity, Argumentation, Stance

Digital Object Identifier 10.4230/OASIcs.LDK.2021.39

Supplementary Material *Software (Source Code)*: `https://github.com/Cwaterschoot/Interactivity_scoring`; archived at `swh:1:dir:f369c7b7343ace35ad1a916e37708dbae8dd3252`

1 Introduction

No shortage exists in regard to online discussions, whether raging on social media or on other websites including those of media outlets. A substantial amount of work has focused on particular aspects of such debates, such as filter bubbles, the purported consequence of personalization in search and recommendation algorithms [17], and echo chambers, clusters of like-minded individuals amplifying their unison reasoning [7]. What has been sparsely studied, however, is how individual messages contribute to the interactivity of an online discussion thread, either towards an echo chamber or balanced discussion.

This paper presents a method for the automatic scoring of a discussion thread in terms of interactivity and argument diversity, as well as for grading each individual post within the thread on the basis of interactive contribution at the time of posting. The starting point of the analysis is a dataset of messages where each sample has been labelled for the argument it presents. The case study in this paper to illustrate the scoring of discussion threads deals with the "Zwarte Piet" (Black Pete) debate in the Netherlands, a topic with clear "pro" sides, i.e. in favour of the figure, and "con" side against the continued existence of "Zwarte Piet".

First, the literature on online discussions, echo chambers and argument diversity is discussed. Then, the scoring methodology is unpacked. The paper ends by discussing the methodology, limitations and what to focus on in future research.

2 Background

Echo chambers and social media is a much discussed topic that has received ample attention from different perspectives, whether political, academic or from the media. An echo chamber is understood to be an enclosed, discursive space, online or based on other forms of media, which amplifies the uniform message encapsulated within. This process magnifies the shared opinion within the cluster while insulating it from rebuttal, creating an environment of positive feedback loops [11].

Previous research tends to agree that echo effects exist on social media platforms, even though the concept remains contested [7, 21, 5]. A possible cause for such an echo effect is the fact that social media users have the tendency to discuss matters with like-minded individuals [5]. It has been concluded that this restricted debate increases polarization [1, 20]. However, others have criticised single media studies for echo chamber detection as it does not take into account the "multiple media environment" that we find ourselves in today [6].

The notion of an echo chamber is seen as disadvantageous by dominant conceptions about democracy as well as by stakeholders in media and moderators. Discourse with those holding differing opinions increases understanding of the subject matter and tolerance for those who disagree [16]. This paper aims to contribute to the development of information systems dealing with online discourse, by mapping interactivity of polarized debates.

The automated classification of echo chambers is not a much discussed topic, even though studies have focused on the subject, particularly in the field of politics. One study has outlined that homophily of social media feeds can be determined across groups by assigning users to either Democrats or Republicans [4]. Furthermore, network analysis has shown the online clustering of communities holding similar views regarding climate change [21].

The current model aims to fill the gap and complement the research on echo chamber detection in pro/con-discussions by implementing domain-unspecific calculations based on annotated data, meaning any labelled data can be used, regardless of the debate statement. The unit of analysis is the thread. Such discussions can either be balanced in terms of argumentation or skewed to one perspective. A second indicator is calculated at the message level, as every individual reply in a thread receives a contribution score.

From here on out, an *echo chamber* will refer to a thread in which the argumentative position presented in the parent message – the contribution starting the thread to which others have replied – is continued throughout the thread, per calculation. The opposite, in which the contrasting argumentative camp, whether pro or con, is the dominant presence in the thread, will be called an *opposition flood*. Equal presence of pro and con messaging results in a *balanced* discussion. A thread can be interpreted as a string of messages portraying an argument belonging to either the pro or con camp where all replies comment on the parent message. Simplified examples are as follows in the form $\{firstpost \rightarrow replypost \rightarrow replypost \rightarrow ...\}$:

$$Echo\ chamber := X_{pro} \rightarrow Y_{pro} \rightarrow X_{pro} \rightarrow X_{pro}$$
$$Opposition\ flood := X_{pro} \rightarrow Z_{con} \rightarrow L_{con} \rightarrow M_{con}$$
$$Balanced := X_{pro} \rightarrow Z_{con} \rightarrow Y_{pro} \rightarrow M_{con}$$

2.1 Case study

To illustrate the approach, an annotated dataset containing online threads discussing the controversial blackface figure of Black Pete in the Netherlands was created. This discussion has a clear pro/con divide. Those in favour of the figure, a component of the Dutch Sinterklaas festivities, argue that Black Pete ought to remain as it was celebrated throughout

the last decades. The camp opposing the festivities assert that the character is a racist stereotype portraying people of colour and should not be celebrated. This debate ought to be seen more broadly in the discussion on racism in Dutch society [2]. These threads were collected from Twitter (using the keyword "Zwarte Piet"), Reddit, by scraping the subreddit r/thenetherlands with "Zwarte Piet", and finally Gab, also scraped using the hashtag "zwartepiet" (Table 1).

Table 1 Threads and messages included, sorted by platform.

Platform	Total Threads	Total Messages
Twitter	21	125
Reddit	7	39
Gab	7	22

Manual labelling with regard to the included arguments was performed, based on the outline presented in previous research (see e.g. [18, 2, 10, 9] and Table 2). Stance labelling of social media data is a challenging task and therefore, it is done at the level of argumentation presented in the literature [13, 12].

Table 2 Arguments (Labels) in the Zwarte Piet discussion.

Level1 (l1)	Level2 (l2)
Pro	Dutch tradition, Christian tradition, Innocent, Intention, Pre-christian, Oriental
Con	Racial stereotype: historical, Racial stereotype: contemporary

Each post in the data was labelled for the dominant argument (level2) that it presents in regard to the "Zwarte Piet" discussion (Table 2). These labels have been derived from the extensive literature outlining this particular debate in The Netherlands. To test whether such argumentation can be clearly detected in online contributions, multiple annotators were employed to label all gathered posts. The annotators were familiarized with the discussion and arguments using the existing literature (see e.g. [18, 2, 10, 9]). Furthermore, a sheet with all possible labels alongside a brief explanation was provided to guide the labelling process. A Krippendorff's alpha of 0.745 was calculated, indicating that inter-rater agreement exists.

3 Methodology

We propose a calculation method for estimating indicators of interactivity in threads. A first indicator applies to the thread level; a second indicator relates to single messages.

The model created in this paper makes certain assumptions in order to compute interactivity. First, each post contributes at least one argument in the discussion. Second, each argument can be assigned to a position in the discussion, whether it be "pro" or "con". Additionally, it is assumed that the more an argument is repeated, the smaller the contribution a new repetition will make in terms of diversity/interactivity on the individual message level. However, when calculating the state of the thread as a whole, a new repetition will weigh greater towards the extremes of echo chamber/opposition flood, i.e. constant repeating of identical reasoning will result in an echo chamber or opposition flooding faster.

3.1 Thread Interactivity Score

The thread as a whole receives a single score based on the interactivity and diversity detected in the posts. This real-valued indicator provides information on whether the presented collection of arguments constitutes an echo chamber, opposition flood or a balanced discussion. To compute the overall thread interactivity score, each message receives a cumulative log operator, which increases as an identical argument is repeated within the thread. Using this factor, repetition of a single reasoning weighs heavier towards the extremes, either echo chamber or opposition flood.

Calculating the log operator for both the echo and opposition scores requires the cumulative count of the argument (denoted as j) in each message at that point in time. Simply put, this variable equals the nth iteration of the particular argument represented in the sample at the order given in the data. To calculate the actual log operator, $log_{10}(j)$ is substracted. Dividing the log operator by the total number of messages in the thread (N) results in the message share. Per the assumptions, each argument can be assigned to either the "pro" or "con" side, which is notated as $l1$ of an argument, the deciding factor whether the share is negative or positive (denoted as multiplication by -1). The specific argument as presented in the case study is decoded as $l2$. The Thread Interactivity Score (TIS) is sum of all shares in thread T. An exception exists for replies where the specific argument is identical to the parent message. In this case, the share is multiplied by a weight and added to the parent message share that is not weighed down, with the result that a parent repetition impacts the echo score to a larger degree.

$$Share_i \begin{cases} \frac{j(x_i)-1-log_{10}(j(x_i)-1)}{N} * (-w) + \frac{1}{N} & \text{if } l2(x_i) = l2(x_0) \\ \frac{j(x_i)-log_{10}(j(x_i)}{N} & \text{if } l2(x_i) \neq l2(x_0) \wedge l1(x_i) \neq l1(x_0) \\ \frac{j(x_i)-log_{10}(j(x_i)}{N} * (-1) & \text{if } l2(x_i) \neq l2(x_0) \wedge l1(x_i) = l1(x_0) \\ 0 & \text{if } i = 1 \end{cases} \tag{1}$$

$$TIS_T = \sum_{i=1}^{N} share_i$$

A perfectly balanced discussion will have a TIS of 0, indicating that both the echo share and opposition are equal. An echo chamber is defined as a thread with a TIS below -0.5. Dipping below this threshold means that the share of echo posts is more than double that of the opposition posts. Threads with a TIS above 0.5 are overflooded with opposition messaging.

The opposition score is defined as the sum of shares of all messages from the opposite side of the parent argument on $level1$ (l1), while the echo score is the result of summing the shares in absolute value of all messages where $level1$ equals that of the parent.

To detect when a thread turns into an echo chamber or opposition flood, the TIS is calculated at each new posting in an iterative manner. Thus, it combines the log operator from the TIS with a time-dependent factor. This approach might enable future research to study trends in online discussions in regard to echo chamber prediction. The result is a matrix of message shares, calculated at each new posting in the thread at that point in time. Dynamic scoring follows the TIS equation(1) in which thread size N equals message index i at the point of calculation.

3.2 Message Interactivity Contribution

Alongside the indicators calculated at the thread level, individual posts receive a diversity score representing the extent to which this post at the time of posting contributed to the thread in terms of interactivity. Simply put, if the new post presents an argument that

has not been part of the discussion, it contributes more to the thread compared to when perspectives are repeated. Subsequent repetition of identical arguments are downgraded by the individual log operator, which decreases the more an already presented argument is added. The message contribution of reply i is calculated as follows:

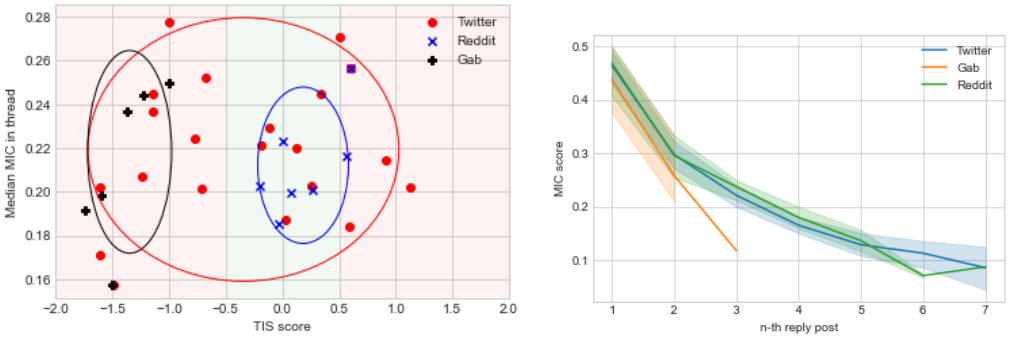
$$MIC_i \begin{cases} \frac{1-log_{10}(j(x_i))}{i} * w^{-1} & \text{if } l2(x_i) = l2(x_0) \\ \frac{(1-log_{10}(j(x_i))}{i} & \text{if } l2(x_i) \neq l2(x_0) \\ 0 & \text{if } i = 0 \end{cases}$$

To derive this MIC indicator, the message share at that point in time is calculated using the individual log operator, which decreases if an argument was already prevalent in the discussion. This share equals one minus the log of the cumulative count of the argument, i.e. j, divided by the number of arguments in the thread at the point in time of the message (i). The first post of a thread always receives MIC equal to zero, as it is not a reply and due to the thread score remaining zero at that point in time. When the parent argument is repeated, the contribution is downgraded by the inverse of the weight. Large MIC values indicate greater contribution to the argument diversity within the thread. Following Equation 3.2, the MIC in a thread converges to zero as the thread size grows.

To determine whether a message is an interactive contribution to the thread in terms of argument diversity, the current MIC value of post i is compared to that one of the previous post $i-1$. Replies with a greater MIC score than the previous post are deemed interactive contributions. In case the first reply post contains identical argumentation to the original post, it cannot be seen as a contribution in terms of interactivity.

4 Results

The first obtained indicator is the Thread Interactivity Score (TIS), the overall score as a whole, plotted alongside the median MIC score in the thread (Figure 1a). TIS informs you whether the thread is an echo chamber, balanced debate or opposition flood. Balanced discussion is found when the TIS falls within the interval $[-0.5, 0.5]$, indicating a somewhat equal distribution of arguments. Threads with a score below -0.5 are deemed echo chambers, above 0.5 as opposition floods where the parent argument is overflooded by opposing messages. For this particular illustration, the weight for *punishing* repetition of the parent post was kept at 1.1.



(a) Dynamic TIS.

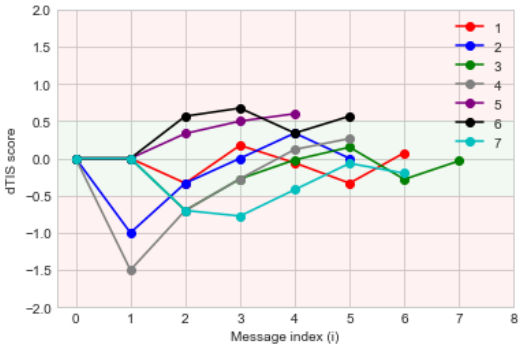
(b) Average MIC at the n-th reply, 95% ci.

Figure 1 Dynamic TIS & MIC scores, Black Pete case study, by platform.

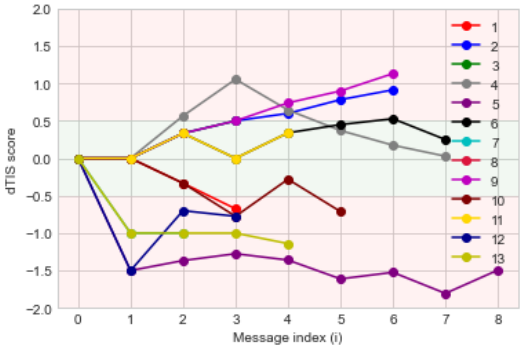
The three online platforms showcase different characteristics in regard to overall thread status, at least in this dataset (Figure 1a). Gab appears to exclusively host echo chambers, confirming previous research [14]. The "Zwarte Piet" discussion on Reddit, however, results in balanced discussion with the exception of two threads. Finally, the TIS result indicates that one finds variability on Twitter regarding the thread status, with both echo chambers, balanced discussion and opposition flooding found in this dataset (Figure 1a). That being said, the 21 Twitter threads plotted here do collectively shift slightly towards echo chambers.

The dynamic TIS (dTIS) informs how a thread developed in terms of argument diversity and interactivity. Figure 2 visualizes threads from all included platforms. One can infer from the dTIS when a thread becomes an echo chamber (dipping below -0.5) or if it returns into the green zone, indicating a balanced discussion.

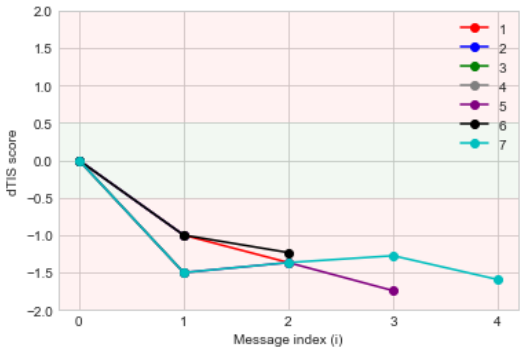
Figure 2 indicates that Gab lacks any argumentation from one side of the aisle, resulting in direct echo chambers. Secondly, threads on Reddit bounce back towards balanced discussion even when the first replies pull the thread towards an echo chamber. Furthermore, the variability in thread structure on Twitter are once again visible. Some discussions are echo chambers from the first reply onwards, never experiencing opposite messaging (e.g. thread 5, thread 13), others bounce back and forth between balanced and echo chamber (thread 10). On the other side of the spectrum, threads steadily grow towards opposition flood, meaning that every new reply to the thread argued against the parent message (thread 2, thread 9).



(a) Reddit, n=7.



(b) Twitter, first 14 threads.



(c) Gab, n=7.

Figure 2 Dynamic TIS scores per platform, balanced discussion in $[-0.5, 0.5]$.

Moving on from the thread scoring, the MIC score reflects how much the post in question contributed to the argument diversity at that point in time. Figure 1b summarizes this scoring by averaging the MIC score at each subsequent reply across platforms in the dataset.

In the case of Gab, where maximum thread size is four, it is clear that, due to the absence of diversity in arguments, replies quickly diminish in terms of contribution. Due to the linear MIC decline in the scraped threads, no reply posts can be deemed beneficial contributions in terms of argument diversity.

However, this cannot be said for the threads scraped from Twitter and Reddit (Figure 1b). The decline in message contribution is less steep compared to Gab. Furthermore, on Reddit, 14 replies were deemed interactive, meaning that the MIC was larger than the previous message. In the case of Twitter, 30 replies were found to be interactive, accounting for about a quarter of included comments.

In the case of the "Zwarte Piet" dataset used for this calculation, one could infer that the most diverse debate in terms of argumentation is found on Reddit, due to the fact that a larger share of comments are deemed interactive, combined with the absence of a field dominated by echo chambers. However, this dataset is limited both in scope and size. While these indicators can be used to explore online discussions, in this instance it is a mere illustration of the calculation and variables.

5 Discussion & conclusion

This short paper presented a calculation procedure for two metrics for estimating echo chamber effects in online discussion threads. The case study, focusing on the "Zwarte Piet" discussion in the Netherlands, illustrated how the debate exists on different online platforms. Threads belonging to the right-wing network Gab exclusively fall into the echo chamber category, in line with the literature [14, 23]. In this specific dataset, the discussion around the "Zwarte Piet" figure on subreddit r/thenetherlands falls mostly within the balanced category. Previous research put forward varied results in terms of echo chambers on Reddit depending on the subreddit in question [15]. Concerning the valuation of replies, the Reddit threads hold a larger share of interactive comments compared to Twitter. Furthermore, the discussion on Twitter experiences wide variability with a slight collective shift towards echo chambers. This divergence in thread status is reflected in previous research on the social media platform, as studies report a variety in results regarding bias and homophily on Twitter feeds [3, 21, 19]. Political studies as well as studies focussing on climate change tend to point towards echo effects on Twitter [8, 22].

Posts deemed interactive by MIC calculation can be valuable for stakeholders. Journalists and moderators aim to have engaging forum discussions on their platform with a large number of participants. Academics might look at interactive posts to map out discussions, understand echo chambers and what effects they have on deliberative debate.

While the discussed indicators do confirm previous research, the approach is not without its limitations. First, for the approach to provide valid and qualitatively sound scoring, an annotated dataset is needed. This data ought to be labelled for the specific argument or debate stance put forward in the message. Without substantiated labelling, the scoring loses value and interpretability. However, as illustrated by the case study, when threads are well-annotated, the scoring yields understandable results.

The TIS and MIC scoring informs about the status of a thread and contribution of a message in the discussion in terms of argument diversity and interaction across argumentative camps. However, what it lacks is any indication on the quality of the interaction taking place. Understandably, a wide variety exists in terms of constructive communication among posters on internet platforms and social media. This approach operates at the coarse pro/con and basic argumentative levels, ignoring further depth of the communicative discourse.

Further research is needed to address these limitations. The current study is small in scope and size. A larger case is needed to rigidly map out echo chambers on online platforms with the goal of being independent of topic, platform or language. Different weights for parent argument repetition ought to be included as well in order to pinpoint the effect. Additionally, the concept of interaction in online discussion needs to be unpacked in further detail by developing estimators for qualitative features of interaction. By introducing gradation in terms of discursive quality in the process of valuating reply contribution, the depth of such interaction can be included. Studies to come will pinpoint just that aspect of online threads in order to fill this gap. Moreover, future work will focus on the automatic labelling of online posts in regard to presented argumentation. While in this proof-of-concept study this was done manually, the automatic annotation of pro- and con-statements allows for a computational pipeline for echo chamber detection from the ground up. Upcoming research will address just that, using the "Zwarte Piet" case as well as other discussion cases to include broader topics that do not showcase such strong binary distinction between pro- and con-groups.

The concrete necessity to better outline and understand online discourse and echo chambers becomes more urgent as social media and other online platforms acquire dominance in societal conversation. As this trend progresses, so does the need for research to follow that path and develop automated methods that help detecting adverse and toxic discourse and communication. The presented calculation aims to contribute to this challenge by expanding the computational possibilities for forum and discussion moderation.

References

1 Alan I. Abramowitz and Kyle L. Saunders. Is polarization a myth? *Journal of Politics*, 70(2):542–555, 2008. doi:10.1017/S0022381608080493.

2 Markus Balkenhol. Zwarte Piet, racisme en emoties. *Waardenwerk*, 62/63:36–46, 2015.

3 Axel Bruns. Echo Chamber? What Echo Chamber? Reviewing the Evidence. *Future of Journalism 2017*, pages 1–11, 2017. URL: http://snurb.info/files/2017/EchoChamber.pdf.

4 Elanor Colleoni, Alessandro Rozza, and Adam Arvidsson. Echo Chamber or Public Sphere? Predicting Political Orientation and Measuring Political Homophily in Twitter Using Big Data. *Journal of Communication*, 64(2):317–332, 2014. doi:10.1111/jcom.12084.

5 Siying DU and Steve Gregory. The Echo Chamber Effect in Twitter: does community polarization increase? *Studies in Computational Intelligence*, 693:V–VII, 2017.

6 Elizabeth Dubois and Grant Blank. The echo chamber is overstated: the moderating effect of political interest and diverse media. *Information Communication and Society*, 21(5):729–745, 2018. doi:10.1080/1369118X.2018.1428656.

7 Seth Flaxman, Sharad Goel, and Justin M. Rao. Filter bubbles, echo chambers, and online news consumption. *Public Opinion Quarterly*, 80(Special Issue 1):298–320, 2016.

8 Kiran Garimella, Gianmarco De Francisci Morales, Aristides Gionis, and Michael Mathioudakis. Political discourse on social media: Echo chambers, gatekeepers, and the price of bipartisanship. *WWW '18: Proceedings of the 2018 World Wide Web Conference*, pages 913—922, 2018.

9 John Helsloot. Zwarte Piet and Cultural Aphasia in the Netherlands. *Quotidian: journal for the study of everyday life*, 3:1–20, 2012.

10 John Helsloot. *Contesting Ambiguity: The Black Peter Mask in Dutch Cultural Heritage*, volume 5 of *The Ritual Year*, pages 124–132. Vytautas Magnus University, 2013. Reporting year: 2013.

11 Kathleen Jamieson and Joseph Capella. *Echo Chamber: Rush Limbaugh and the Conservative Media Establishment*. Oxford University Press, 2008.

12 Dilek Küçük and C. A.N. Fazli. Stance detection: A survey. *ACM Computing Surveys*, 53(1), 2020. doi:10.1145/3369026.

13 Florian Kunneman, Mattijs Lambooij, Albert Wong, Antal Van Den Bosch, and Liesbeth Mollema. Monitoring stance towards vaccination in twitter messages. *BMC Medical Informatics and Decision Making*, 20(1):1–14, 2020. doi:10.1186/s12911-020-1046-y.

14 Lucas Lima, Julio C.S. Reis, Philipe Melo, Fabricio Murai, Leandro Araujo, Pantelis Vikatos, and Fabricio Benevenuto. Inside the right-leaning echo chambers: Characterizing gab, an unmoderated social system. *Proceedings of the 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2018*, pages 515–522, 2018.

15 Richard A. Mills. Pop-up political advocacy communities on reddit.com: SandersForPresident and The Donald. *AI and Society*, 33(1):39–54, 2018. doi:10.1007/s00146-017-0712-9.

16 Diana Mutz and Jeffrey Mondak. The workplace as a context for cross-cutting political discourse. *The Handbook of Discourse Analysis*, 68(1):398–415, 2006. doi:10.1002/9780470753460.ch21.

17 Eli Pariser. *The Filter Bubble: What the Internet is hiding from you.* Penguin Books Ltd, 2011.

18 Heleen Schols. *Keeping things gezellig: Negotiating Dutchness and racism in the struggle over 'Black Pete'.* PhD thesis, Amsterdam Institute for Social Science Research, 2020.

19 Dominic Spohr. Fake news and ideological polarization: Filter bubbles and selective exposure on social media. *Business Information Review*, 34(3):150–160, 2017.

20 Cass R. Sunstein and Adrian Vermeule. Symposium on conspiracy theories: Conspiracy theories: Causes and cures. *Journal of Political Philosophy*, 17(2):202–227, 2009.

21 Hywel T.P. Williams, James R. McMurray, Tim Kurz, and F. Hugo Lambert. Network analysis reveals open forums and echo chambers in social media discussions of climate change. *Global Environmental Change*, 32:126–138, 2015.

22 Hywel T.P. Williams, James R. McMurray, Tim Kurz, and F. Hugo Lambert. Network analysis reveals open forums and echo chambers in social media discussions of climate change. *Global Environmental Change*, 32:126–138, 2015.

23 Savvas Zannettou, Barry Bradlyn, and Emiliano De Cristofaro. What is Gab ? A Bastion of Free Speech or an Alt-Right Echo Chamber ? *WWW '18: Companion Proceedings of the The Web Conference*, 2018.

Linking Discourse Marker Inventories

Christian Chiarcos
Applied Computational Linguistics Lab, Goethe University Frankfurt, Germany

Maxim Ionov
Applied Computational Linguistics Lab, Goethe University Frankfurt, Germany

Abstract

The paper describes the first comprehensive edition of machine-readable discourse marker lexicons. Discourse markers such as *and*, *because*, *but*, *though* or *thereafter* are essential communicative signals in human conversation, as they indicate how an utterance relates to its communicative context. As much of this information is implicit or expressed differently in different languages, discourse parsing, context-adequate natural language generation and machine translation are considered particularly challenging aspects of Natural Language Processing. Providing this data in machine-readable, standard-compliant form will thus facilitate such technical tasks, and moreover, allow to explore techniques for translation inference to be applied to this particular group of lexical resources that was previously largely neglected in the context of Linguistic Linked (Open) Data.

2012 ACM Subject Classification Computing methodologies → Discourse, dialogue and pragmatics; Information systems → Graph-based database models

Keywords and phrases discourse processing, discourse markers, linked data, OntoLex, OLiA

Digital Object Identifier 10.4230/OASIcs.LDK.2021.40

Supplementary Material *Software (Source Code)*:
`https://github.com/acoli-repo/rdf4discourse`

Funding This work was funded by the project "Prêt-à-LLOD" within the European Union's Horizon 2020 research and innovation programme under grant agreement no. 825182, as well as the project "Linked Open Dictionaries" (LiODi), funded within the eHumanities program of the German Ministry of Education and Science (BMBF, 2015-2021).

1 Motivation and Background

Natural language does not exist in isolation, but always fulfills a communicative purpose, be it to inform an addressee about a specific state of affairs, to motivate them to perform certain acts, to bond or to interact with them otherwise (e.g., convince the addressee of a certain belief). Much of this information, however, resides outside the scope of classical machine-learning based natural language processing: off-the-shelf NLP tools tended to focus on sentences, their components and the grammatical (and semantic) relations between them. With the rising maturity of solutions for more elementary NLP tasks, the automated processing of pragmatics and discourse information did, however, come back into the focus of the discipline and has been subject to a considerable number of shared tasks in recent years, e.g., the 2016 CoNLL Shared Task on Shallow Discourse Parsing[1], the 2019 Shared Task on Discourse Representation Structure Parsing [1] and others.

Discourse markers are a key to the analysis of discourse structure as they represent explicit (albeit not unambiguous) signals of semantic or pragmatic relations that link an utterance with its communicative context (discourse relations), and this has been explored to synthesize training data [39], and is generally considered to be a fast way to light-weight, practical discourse annotation [9].

[1] See `https://www.conll.org/previous-tasks`.

As the result of the COST Action IS1312 "Structuring Discourse in Multilingual Europe" (TextLink, 2014-2018),[2] a considerable number of multilingual discourse marker lexicons has been produced [40], largely following the model of the German DimLex collection [41], but mapped to the sense inventory of the Penn Discourse Treebank [36]. Building on this work and other discourse marker inventories, this paper describes the publication of an interlinked, multilingual discourse marker lexicons on the basis of machine- rather than human-readable form in accordance with web standards and best practices established in computational lexicography, namely as (Linguistic) Linked Open Data [19] and in conformance to OntoLex-Lemon [20].[3]

Motivation for doing so is two-fold: On the one hand, discourse markers inventory becomes more easily accessible for its potential use by off-the-shelf tools, both in individual data sets and as a multilingual graph. The format of the original data sets has considerable variation, even within the TextLink data set: Even though all TextLink discourse marker lexicons are available as XML using the original DimLex lexicon format as a template, they do not adhere to a consistent schema, many contain language-specific extensions, not all are XML-valid, and certain language editions even went so far to translate the original English element and attribute names into the object language. In the language-specific sense inventories, while all based on the Penn Discourse Treebank [36, PDTB], versions 2.0 or 3.0, we also see a certain degree of variation. As a result, this data, while being unquestionably valuable, cannot be directly applied for any NLP task. A better curated version of this data does exist as part of the "Connective-lex" database [40],[4] but the database provides human-readable information only and without explicit licensing information (i.e., restricted).

On the other hand, we also aimed to create further links between discourse marker lexicons and more general lexical resources already available in OntoLex-Lemon. This includes, for example, the Open Multilingual WordNet[5], linked across different languages by means of the Collaborative Interlingual Index[6] [6]. More relevant for the specific case of discourse marker lexicons may be, however, general bidictionaries as provided, for example, as part of the ACoLi Dictionary Graph [16], a large-scale collection of more than 3000 machine-readable bi-dictionaries in OntoLex-Lemon, covering more than 400 language varieties. By linking with this kind of data, it becomes possible to explore techniques to extrapolate discourse marker inventories for low-resource languages by means of techniques as similarly applied for translation inference [38, 28].

Related research on modelling discourse relations and discourse connectives in RDF and/or as Linked Data exists in the form of a suggested discourse extension [27] of the General Ontology of Linguistic Description [23] employed for discourse parsing [4]. At the time of writing, this resource is no longer publicly available but can be partially reconstructed from associated publications [31]. The FRED machine reading system [26] produces OWL output from an NLP stack that also incorporates an off-the-shelf discourse parser [7], but it does not seem[7] to provide a vocabulary for discourse relations.

---

[2] `http://textlink.ii.metu.edu.tr/`

[3] We acknowledge that other authors established a level of machine-readability in earlier research by providing discourse marker inventories in XML rather than printed form [40]. While this establishes machine-readable syntax, we build and extend this work by establishing machine-readable *semantics*.

[4] `http://connective-lex.info/`

[5] `http://compling.hss.ntu.edu.sg/omw/`

[6] `https://github.com/globalwordnet/ili`

[7] The full vocabulary of the FRED system is not publicly documented. The observations above are insights obtained from example queries.

2 Discourse markers and discourse marker lexicons

Discourse markers, also referred to as discourse cues or discourse connectives, do not constitute a homogenous class of grammatical devices in most languages, but rather involve different aspects of grammar, in particular, if described from a cross-linguistic perspective. Accordingly, what constitutes a discourse marker may be defined differently in different theoretical frameworks and for different languages. In most European languages, prototypical discourse markers include conjunctions (such as English *and*, *but*, *if*, etc.), adverbials (such as *thereafter*, *so*), interjections (e.g., *indeed*), but can also be phrasal expressions (*in order to*). In addition to this, certain uses of punctuation (in written language) can be considered to serve as discourse cues, e.g., commas (as markers of lists or enumerations) or hyphens (as markers of contrast or elaboration). Morphological features may serve as discourse cue as well. In this paper, however, we focus on *lexical* discourse markers, i.e., expressions consisting of one or multiple lexemes.

We follow the Penn Discourse Treebank [36, PDTB] in assuming that discourse markers trigger (or indicate) the discourse relation that connects the (proposition expressed by the) local utterance (ARG2 in PDTB terminology, the utterance that contains the discourse marker) with an element in the context (ARG1 in PDTB terminology), so that the type of discourse relation is taken to be the sense of this relation. A discourse marker lexicon is then defined as a dictionary of discourse markers that minimally provides the form(s) of the discourse marker along with one or multiple discourse relations, as well as additional information, e.g., grammatical features, information about uses of the expression other than as discourse marker, frequency and usage information, provenance. It is to be noted that the discourse relations under consideration should be defined as a closed set with fixed identifiers, e.g., defined by an annotation manual. In particular, occasional, but often unsystematic remarks about uses of adverbs as found in traditional dictionaries (e.g., "adversative") are not sufficient to qualify for a discourse marker inventory.

In that sense, a minimal resource that qualifies as discourse marker lexicon is, for example, an aggregate excerpt from a discourse-annotated corpora that lists discourse markers along with the discourse relations these co-occur with, optionally with frequency information. Provided in machine-readable form, such information is an essential tool in technical challenges such as discourse parsing, natural language understanding and natural language generation, and this is where we see the primary application of the data addressed here.

Designated discourse marker lexicons in this sense have been produced since the 1990s, with early examples represented by Alistair Knott [30] and Stede and Umbach [41]. Knott's discourse marker lexicon is available as an appendix to his PhD thesis, and, effectively, has been represented as a plain list. DimLex, the model of Stede and Umbach, originally applied to data from German and English, has become particularly influential in the context of the TextLink initiative, which led to the creation of a relatively consistent set of multilingual discourse marker lexicons. By "relatively consistent", we mean that data is available in XML formats (inspired by the original DimLex XML format, but with language-specific adaptations), and that their sense information has been normalized against the discourse relation inventory of the Penn Discourse Treebank [36, PDTB]. However, the data is far from uniform, most have TextLink dictionaries have been updated to PDTB 3.0 specifications, but some remain at PDTB 2.0 (and the Czech and French datasets uses their own relation inventories, which we mapped as part of the conversion), and likewise, that there is variation in the XML format(s) being used, so that there is no DTD or schema that all these can be validated against. At the core of our data are the following discourse marker lexicons:

DimLex German [41], CC-BY-NC-SA 4.0; extended to Arabic and Bangla [21]; DimLex-XML, PDTB 3.0 relations.

PDTB English, excerpt from Penn Discourse Treebank guidelines [36]; DimLex-XML, PDTB 3.0 relations.

LICO Italian [24], CC-BY 4.0; modified DimLex-XML, PDTB 2.0/3.0 relations.

CzeDLex Czech, bootstrapped from Prague Discourse Treebank 2.0 [34], CC-BY-NC-SA 4.0; PML-XML, PDiT 2.0 relations [45]

LDM-PT Portuguese [33], CC-BY-NC-SA 4.0; DimLex-based XML, PDTB 3.0 relations.

LexConn French [37]; DimLex-inspired XML, SDRT relations [3].

DisCoDict Dutch [8], CC-BY-NC-SA 4.0; DimLex-XML, PDTB 3.0 relations.

A curated version of this data with extensions, consolidated formats and PDTB 3.0 sense linking is accessible from `http://connective-lex.info/`, but for browsing and search only, not for download. We did consult an older version of this data in a partially consolidated state as available from `https://github.com/discourse-lab/Connective-Lex.info`. Using this as a basis we performed format consolidation and linking to PDTB 2.0 for DimLex, DimLex-Arabic, DimLex-Bangla, and LDM-PT. Note that we went for PDTB 2.0 instead of PDTB 3.0 in order to facilitate interoperability with the OLiA Discourse Extensions. CzeDLex and LexConn were converted from the original sources.

Aside from these discourse marker inventories that represent more or less direct results of TextLink, we also converted the DiscMar inventories for English, Spanish and Catalan by Lausa Alonso y Alemany [25], available from `https://cs.famaf.unc.edu.ar/~laura/shallowdisc4summ/discmar/` (HTML format, own relation set), as well as the discourse marker inventory of the TED-Multilingual Discourse Bank (TED-MDB) corpus [43], available under CC-BY from `https://github.com/MurathanKurfali/Ted-MDB-Annotations` (PDTB annotation format, PDTB 2.0/3.0 relations for English, German, Lithuanian, Polish, Portuguese, Russian and Turkish). For the latter, we provide a converter from PDTB annotation files to DimLex-XML, which can subsequently be applied to other PDTB-based corpora such as for Hindi [35] and Chinese [44] that are currently not covered.

For these data sources, we provide a conversion via DimLex-XML to OntoLex-Lemon, and further, a linking with the PDTB 2.0 ontology previously developed as part of the OLiA Discourse Extensions [13]. On this basis, at least two novel modes for querying the relations between discourse markers become possible:

- discourse marker ↦ PDTB concept ↦ discourse marker (from a given discourse marker, retrieve PDTB-equivalent discourse markers)
- discourse marker ↦ PDTB ontology ↦ discourse marker (use the PDTB ontology for imprecise matches, i.e., more general/more specific senses)

A third querying strategy allows to expand the sense information of a discourse marker lexicon, i.e., to apply it for annotation or disambiguation tasks for annotation schemes other than PDTB:

- discourse marker ↦ PDTB ontology ↦ OLiA Discourse Extensions ↦ discourse relations according to other schemes

Although the work described here is grounded on data sets that have been in existence before, with this paper, we describe the first application of Linked Data principles to this kind of data. As a result, improved means of querying local and web-accessible reference data become available only as a result of the conversion and linking activities described in

this paper. As we rely on the general accuracy of the original data, we do not evaluate qualitative performance; instead, subject of evaluation is the capability to formulate and execute these four types of cross-resource queries.

3 From DimLex-XML to OntoLex-Lemon

For the conversion of discourse marker lexicons, we focus on DimLex-XML. Most data sets required considerable pre-processing in order to either consolidate or to produce DimLex-XML, but as a first step, our processing aims to establish a DimLex-conformant level of representation to start with, either from the original XML discourse marker lexicon (DimLex-XML or otherwise), directly from PDTB-style annotations (for TED-MDB), or from a proprietary format (DiscMar).

Using an XSLT 2.0 script, the resulting DimLex file is then transformed to OntoLex-Lemon in Turtle. For reasons of space we do not provide an in-depth description of OntoLex-Lemon, but refer to the original specification [20]. The most relevant OntoLex elements in our context are:

ontolex:LexicalEntry unit of analysis of the lexicon, groups together one or more forms and one or more senses.
ontolex:Form string form of a lexical entry, e.g., written representation.
ontolex:LexicalSense word sense of a particular lexical entry.

Furthermore, a sense can be linked with an externally defined ontological entity by means of `ontolex:reference`. We will use this mechanism to link (lexical entries/senses of) discourse markers with discourse relations.

The OntoLex converter consists of two principal types of conversions, format-specific and generic. Format-specific transformations include:

- For every `entry` element,
  - create an instance of `ontolex:LexicalEnty`.
- For every `orth` element, attach to the entry an `ontolex:Form` by means of either
  - `ontolex:lexicalForm` (for DimLex dialects that do not define canonical forms), or
  - `ontolex:canonicalForm` (for every `orth` element with attribute `canonical="1"`), or
  - `ontolex:otherForm` (for every other `orth` element in a DimLex dialect that defines canonical forms), and
  - assign this form a `ontolex:writtenRep` that contains a language-typed string.
- For every `pdtb3_relation`,
  - create an `ontolex:Sense`, and
  - link it to the lexical entry by means of `ontolex:isSenseOf`.

All other components of the format are converted by generic transformations. For every element that contains (descendants with) attributes or CDATA content:

- identify the element created by the parent element as subject,
- create a property in the `dimlex:` namespace that takes the local name of the current element, and
- assign this property an object, this is either
  - the enclosed text as untyped literal (if the element carries neither attributes nor child elements), or
  - a blank node that serves as subject for properties generated from attributes or (recursively) from child elements.

  XML attributes are likewise preserved as datatype properties with untyped string values.

As namespace for the `dimlex:` elements, we resort to the DimLex DTD `https://github.com/discourse-lab/dimlex/blob/master/DimLex.dtd#`. However, note that several DimLex-style data sets do not validate against this DTD. In this way, we establish core data structures of OntoLex-Lemon, but perform a generic and lossless transformation of XML data structure. This converter is thus capable to support any DimLex dialect and (with minor modifications) related formats. In particular, all resource-specific extensions can be preserved.

The following listing shows the first entry of the German DimLex (with minor omissions):

```
<dimlex>
   <entry id="k1" word="aber">
      <orths>
         <orth type="cont" canonical="1" onr="k1o1">
            <part type="single">aber</part>
         </orth>
      </orths>
      <non_conn_reading>
         <example type="ADV" tfreq="940">aber und abermals</example>
         <example type="ADV">Du bist aber fies!</example>
      </non_conn_reading>
      <syn>
         <cat>konnadv</cat>
         <ordering>
            <ante>0</ante>
            <post>1</post>
            <insert>0</insert>
         </ordering>
         <sem>
            <pdtb3_relation sense="concession-arg2-as-denier" freq="7" anno_N="18"/>
         </sem>
      </syn>
   </entry>
   ...
</dimlex>
```

For all XML elements and attributes below `entry`, the generated Turtle preserves this information faithfully (likewise simplified), in the same order and the same embedding depth:

```
  :k1_aber a ontolex:LexicalEntry;
      ontolex:canonicalForm [ ontolex:writtenRep "aber"@de; dimlex:type "cont";
                              dimlex:onr "k1o1"; dimlex:type "single"];
    dimlex:syn [
      dimlex:cat "konnadv";
      dimlex:ordering [ dimlex:ante "0"; dimlex:post "1"; dimlex:insert "0" ];
      dimlex:sem [
        dimlex:pdtb3_relation [ dimlex:sense "concession-arg2-as-denier";
                                dimlex:freq "7"; dimlex:anno_N "18";
        a ontolex:LexicalSense; ontolex:isSenseOf :k1_aber ] ] .
```

In addition to the OntoLex properties, additional information from the XML format is provided by properties from the DimLex namespace that mirror the original structure of the original XML file. Note that in this way, all information of a DimLex entry can be captured in the RDF graph, but only as far as hierarchy and structure are concerned. The order of elements in the XML is lost in the graph, but also not deemed to be essential for subsequent processing.

A disadvantage of this modelling strategy is that (unless all discourse marker inventories validate against the same schema or DTD – which the publicly available data does not) the resulting DimLex vocabulary is open: Every DimLex-XML dialect can introduce novel datatype and object properties, so that it is not possible to provide an exhaustive class diagram of the DimLex vocabulary in RDF. But we capture the information about basic OntoLex data structures in an interoperable way.

4 Linking with the OLiA Discourse Extensions

The Ontologies of Linguistic Annotation [12, OLiA] have been developed to formalize annotation schemes and to link them with reference concepts, originally primarily for corpora with morphosyntactic and syntactic annotation, with regard to which OLiA covers more than 100 languages at the time of writing,[8] but also extended for pragmatic phenomena such as coreference, information structure, discourse structure and discourse relations. These OLiA Discourse Extensions [13] reside in a separate branch of the OLiA ontologies.[9] As they are still considered experimental, but with increasing maturity, they are about to be integrated with the OLiA.

In its conception, OLiA aimed to address what could be called the "standardization gap" of linguistic annotation. That means that a consistent and homogeneous standardization of linguistic annotation would either have to be reductionistic and neglect language specific characteristics (cf. Universal Dependencies tagset), or constantly grow in complexity with every new language added to it (cf. the evolution of morphosyntactic guidelines from EAGLES to MULTEXT-EAST) [14].

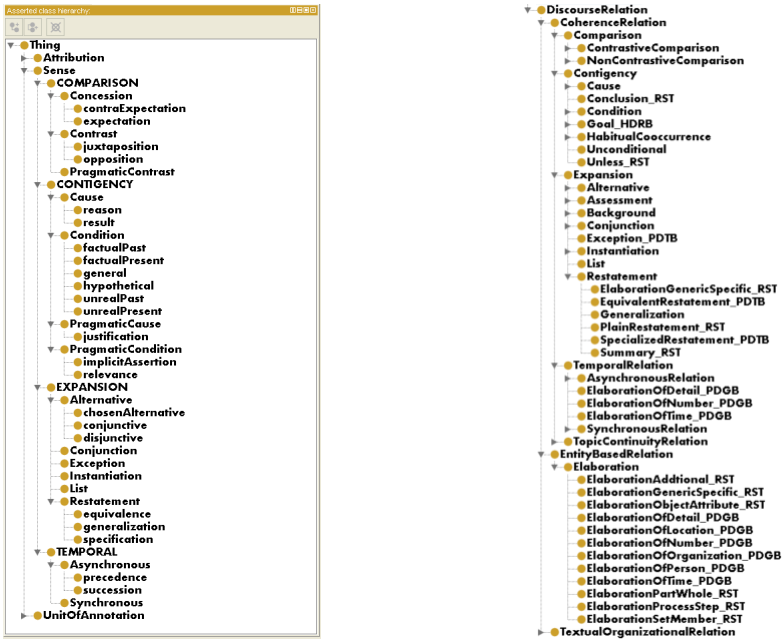
In order to avoid these problems, OLiA introduces an architecture of modular ontologies, formalized in OWL2/DL, to address and to distinguish the different aspects of

- defining concepts and tags relevant for the annotation of a language or a particular language resource (OLiA Annotation Models),
- identifying and defining conventionally used terms (OLiA Reference Model),
- interpreting annotation concepts against reference concepts (OLiA Linking Models, defining `rdfs:subClassOf` relationships between annotation model concepts and reference model concepts), and
- grounding conventionally used terms in reference concepts (external terminology repositories, linking defined as `rdfs:subClassOf` relationships between reference model concepts and externally defined concepts)

With multiple, distinct, but interlinked ontologies, published under CC-BY 3.0 and available under persistent, resolvable URIs, OLiA represents a prototypical application of Linked Data principles to leverage several distributed terminology repositories, and it became subsequently increasingly important as a terminology repository since the conception of the LLOD cloud in 2010, where it represents the central terminology hub for annotation terminology. As of 2020, OLiA is linked with a great number of external terminology repositories, including ISOcat, GOLD, the CLARIN Concept Registry, lexinfo, the Universal Dependency guidelines, the Unimorph guidelines, etc. [15], and it is developed as an open source project under `https://github.com/acoli-repo/olia`.

With the OLiA Discourse Extensions, the approach of modular ontologies and the application of Linguistic Linked Open Data principles to facilitate language resource interoperability can also been extended to the annotation of discourse relations and other aspects of pragmatics. As far as discourse relations are concerned, the OLiA Discourse Extensions cover five annotation schemes based on theoretical work on discourse relations [32, 30, 11, 42, 36] The discourse marker inventories described here are linked to the original PDTB ontology as part of the OLiA Discourse Extensions [13] and available under CC-BY 3.0 from `http://purl.org/olia/discourse/discourse.PDTB.owl`.

[8] `http://purl.org/olia`
[9] `http://purl.org/olia/discourse`



(a) PDTB ontology as provided by the OLiA Discourse Extensions, visualized using Protégé.

(b) OLiA discourse model: Reference concepts for the OLiA Discourse Extensions.

Figure 1 PDTB ontology and OLiA Discourse Extensions.

The PDTB ontology is summarized in Fig. 1a. We focus on the `pdtb:Sense` branch alone, where PDTB asserts the existence of four major types of discourse relations, contrastive relations (COMPARISON), causal relations (CONTIGENCY), temporal relations (TEMPORAL) and additive relations (EXPANSION), with two levels of further refinement. In real-world annotation, an annotator may decide to assign a discourse marker the most specific relation they find in that taxonomy (e.g., *reason*), but likewise, a more abstract relation if none of the subclasses match precisely or seem to be equally applicable (e.g., *Cause*, or even *CONTIGENCY*). We take this to be an implicative hierarchy, i.e., that any more specific discourse relation automatically entails the applicability of a more generic one – albeit this kind of reasoning seems to be rarely applied in current PDTB practice. Instead, the hierarchy has been exploited for evaluating the performance of discourse parsing, where accuracy can be evaluated against different levels of granularity, ranging from top-level (4 discourse relations plus entity relations and no relation) over second-level relations (15 discourse relations) to the full inventory. The discourse marker inventories are linked with the PDTB ontology by means of a simple SPARQL update: If the label of a discourse relation of the PDTB ontology matches the literal value of `dimlex:sense`, insert an `ontolex:reference`, i.e., link the PDTB ontology as an external ontology:

```
INSERT { ?dimlex_relation ontolex:reference ?pdtb_sense. }
WHERE  { ?dimlex_relation dimlex:sense ?label.
         ?pdtb_sense (rdfs:label|skos:altLabel) ?sense_label.
         FILTER(lcase(?label)=lcase(?sense_label))       };
```

For the linked version of the data set, we perform an additional pruning step and omit all properties from the `dimlex:` namespace, i.e., aspects of the original XML content that have, so far, not been interpreted into machine-readable information. (Remember that the

Table 1 Statistics and accessibility information for discourse marker inventories, PDTB-linked, OntoLex-Lemon edition.

lan-guage	dataset http://purl.org/acoli/dimlex/...	license	PDTB links	markers (canonical)	granu-larity
ar	.../ar/arabic.ttl	t.b.d.	505	505	14
bn	.../bn/dimlex-bangla.ttl	CC-BY-NC-SA 4.0	107	122 (101)	16
ca	.../ca/discmar.ca.ttl	CC-BY-NC 3.0	97	93	5
cs	.../cs/czedlex0.6.ttl	CC-BY-NC-SA 4.0	1883	1459 (204)	20
de	.../de/DimLex.ttl	CC-BY-NC-SA 4.0	411	763 (274)	18
de	.../de/ted-mdb-german.ttl	CC-BY 4.0	27	31	15
en	.../en/discmar.en.ttl	CC-BY-NC 3.0	90	98	5
en	.../en/pdtb2.ttl	CC-BY-NC-SA 4.0	535	186 (92)	21
en	.../en/ted-mdb-english.ttl	CC-BY 4.0	23	24	11
es	.../es/discmar.es.ttl	CC-BY-NC 3.0	93	97	5
fr	.../fr/lexconn.ttl	CC-BY-NC 3.0	416	603	13
it	.../it/LICO-v.1.0.ttl	CC-BY 4.0	174	204	19
lt	.../lt/ted-mdb-lithuanian.ttl	CC-BY 4.0	27	24	13
nl	.../nl/discodict.ttl	CC-BY-NC-SA 4.0	244	473 (207)	21
pl	.../pl/ted-mdb-polish.ttl	CC-BY 4.0	4	12	3
pt	.../pt/LDM-v.1.3.ttl	CC-BY-NC-SA 4.0	663	254	22
pt	.../pt/ted-mdb-portuguese.ttl	CC-BY 4.0	21	22	9
ru	.../ru/ted-mdb-russian.ttl	CC-BY 4.0	21	21	11
tr	.../tr/ted-mdb-turkish.ttl	CC-BY 4.0	28	31	11

`dimlex` namespace is merely a placeholder for generic XML information that has not found an interpretation against OntoLex or another RDF vocabulary.) However, the original RDF data is preserved and can be consulted for future extensions.

Table 1 gives an overview over the linking statistics and also provides the persistent URIs for the respective linked data sets. Note that these URIs resolve, and that machine-readable license information is provided, so that the result of conversion and linking represents fully qualified Linguistic Linked (Open) Data.

All resulting data is available under the respective original license from our GitHub repository (`https://github.com/acoli-repo/rdf4discourse/tree/master/discourse-markers/linked`). After conversion and linking, the resulting data has been enriched with machine-readable metadata about the respective license (`dct:license`), and the location of the original data (`dcr:source`). Human-readable details on attribution are provided as `rdf:comment` of the `lime:Lexicon` element that represents the respective discourse marker inventory. Note that not all data sets have an explicit license statement. This includes LexConn, DiscMar and Arabic. As for the first three, the information contained in them corresponds *exactly* to a respective appendix of the accompanying documentation [36, 37, 2]. We consider this as unproblematic in terms of copyright, as the discourse marker inventories created on this basis represent collections of (fully attributed) non-literal quotations. In order to preserve the intended band-width of scientific citations, we assert a CC-BY-NC 3.0 license for these, using the original literature references as attributions.[10] The copyright situation of the Arabic discourse marker lexicon is still unresolved, full attribution is provided, but in case of complaints, it will be withdrawn from the public release.

[10] We adopt CC-BY-NC 3.0 instead of CC-BY-NC 4.0 as the 3.0 BY clause allows authors to enforce the use of a specific title, i.e., a particular form of citation, rather than alternative means of attribution (e.g., by publication URI).

5 Querying

As mentioned before, our evaluation consists of demonstrating the capability to query discourse marker inventories in combination with discourse relation inventories, both the PTDB ontology and the OLiA Discourse Extensions.

With discourse marker inventories, sense definitions and annotation schemes interconnected by means of Linked Data technology, it now becomes possible to traverse the paths in a graph, e.g., in order to retrieve translations or alternative lexicalizations of discourse markers. Note that this functionality is currently not provided by the Connective-Lex database [40], so that this is a novel functionality. The following query retrieves an English word from the DiscMar inventory and its PDTB sense:[11]

```
PREFIX ontolex: <http://www.w3.org/ns/lemon/ontolex#>
PREFIX pdtb: <http://purl.org/olia/discourse/discourse.PDTB.owl#>
PREFIX rst: <http://purl.org/olia/discourse/discourse.RST.owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT distinct ?en ?pdtb
FROM <http://purl.org/acoli/dimlex/en/discmar.en.ttl>
WHERE {
  ?form ontolex:writtenRep ?en.     filter(lang(?en) = "en")
  ?entry (ontolex:lexicalForm|ontolex:canonicalForm) ?form.
  ?sense ontolex:isSenseOf ?entry.
  ?sense ontolex:reference ?pdtb.
} ORDER BY ?en ?pdtb
```

Simplifying the query using property paths and adding a second path with a different language filter, we can now apply this query to derive translations, e.g., between English and German connectives:

```
SELECT distinct ?en ?pdtb ?de  # prefixes are omitted for space reasons
FROM <http://purl.org/acoli/dimlex/en/discmar.en.ttl>
FROM <http://purl.org/acoli/dimlex/de/DimLex.ttl>
WHERE {
  ?pdtb ^ontolex:reference/ontolex:isSenseOf/
        (ontolex:lexicalForm|ontolex:canonicalForm)/
        ontolex:writtenRep ?en.
  filter(lang(?en) = "en")
  ?pdtb ^ontolex:reference/ontolex:isSenseOf/
        (ontolex:lexicalForm|ontolex:canonicalForm)/
        ontolex:writtenRep ?de.
  filter(lang(?de) = "de")
} ORDER BY ?en ?pdtb ?de
```

As a general rule, we would expect that DiscMar results are more coarse-grained than DimLex results. In the SPARQL query, this can be captured by extending the search to retrieve DimLex lexicalization for indirect *subclasses of* DiscMar entries (assuming that the PDTB ontology is loaded in the default graph):

```
SELECT distinct ?en ?pdtb ?de
FROM <http://purl.org/acoli/dimlex/en/discmar.en.ttl>
FROM <http://purl.org/acoli/dimlex/de/DimLex.ttl>
FROM <http://purl.org/olia/discourse/discourse.PDTB.owl>
WHERE {
  ?pdtb rdfs:subClassOf*/^ontolex:reference/ontolex:isSenseOf/
        (ontolex:lexicalForm|ontolex:canonicalForm)/
        ontolex:writtenRep ?en.
  filter(lang(?en) = "en")
  ?pdtb ^ontolex:reference/ontolex:isSenseOf/
        (ontolex:lexicalForm|ontolex:canonicalForm)/
        ontolex:writtenRep ?de.
  filter(lang(?de) = "de")
} ORDER BY ?en ?de
```

[11] This query as well as all subsequent queries can be directly executed with the online SPARQL service provided under `http://www.sparql.org/sparql.html` and have been tested for this purpose. No additional configuration is necessary, as the `FROM` statements indicates the RDF graphs to read from. Alternatively, they can run on *any* local SPARQL end point once the ontologies are loaded.

This query now returns 7,040 different translation pairs for German and English, whereas the former query retrieved only 828 translation pairs. However, note that many of these translations are imprecise because of differences in granularity. As an example, DimLex differentiates between subclasses of causal relations (PDTB *reason* and *result*), whereas DiscMar only identifies clausal relations as PDTB *Cause*. Transitive queries, e.g., along the `rdfs:subClassOf` axis as expressed by the Kleene star in the example, are an efficient way to deal with differences in granularity. An alternative is to enable the RDFS entailment regime in the SPARQL end point. Then, the original query (without the Kleene star) does return a comparable result.[12]

But the linked graph can also be used in other ways. If the PDTB linking model (`http://purl.org/olia/discourse/discourse.PDTB-link.rdf`) is imported into the default graph, we arrive at reference model concepts from the OLiA Discourse Extensions. with other linking and annotation models connected, it becomes possible, then, for example, to "translate" the PDTB relations to RST relations [32], illustrated for the marker *because* according to the PDTB inventory:

```
SELECT distinct ?pdtb ?olia ?rst
# OntoLex and PDTB data
FROM <http://purl.org/acoli/dimlex/en/pdtb2.ttl>
FROM <http://purl.org/olia/discourse/discourse.PDTB.owl>
# OLiA Discourse Extensions
FROM <http://purl.org/olia/discourse/discourse.PDTB-link.rdf>
FROM <http://purl.org/olia/discourse/olia_discourse.owl>
FROM <http://purl.org/olia/discourse/discourse.RST-link.rdf>
FROM <http://purl.org/olia/discourse/discourse.RST.owl>
WHERE {
  ?pdtb rdfs:subClassOf*/^ontolex:reference/ontolex:isSenseOf/
        (ontolex:lexicalForm|ontolex:canonicalForm)/
        ontolex:writtenRep "because"@en.

  ?pdtb rdfs:subClassOf ?olia. # the directly assigned olia senses
  FILTER(contains(str(?olia),"olia_discourse"))

  ?rst rdfs:subClassOf+ ?olia. # RST subsenses
  FILTER(contains(str(?rst),"discourse.RST"))
} ORDER BY ?pdtb ?rst
```

This query returns 11 possible RST relations and also gives information about the path that connects them with the original definition:

pdtb	olia	rst
pdtb:Cause	olia_discourse:Cause	rst:Evidence
pdtb:Cause	olia_discourse:Cause	rst:Justify
pdtb:Cause	olia_discourse:Cause	rst:Motivation
pdtb:Cause	olia_discourse:Cause	rst:NonVolitionalCause
pdtb:Cause	olia_discourse:Cause	rst:NonVolitionalResult
pdtb:Cause	olia_discourse:Cause	rst:Purpose
pdtb:Cause	olia_discourse:Cause	rst:VolitionalCause
pdtb:Cause	olia_discourse:Cause	rst:VolitionalResult
pdtb:Condition	olia_discourse:Condition	rst:Condition
pdtb:Condition	olia_discourse:Condition	rst:Enablement
pdtb:Condition	olia_discourse:Condition	rst:Means

[12] The difference is in the binding for the variable `?pdtb`: In the query with the Kleene star, we retrieve the more specific sense, as expressed in DimLex. In the query without the Kleene star but RDFS entailment enabled, we retrieve the more general sense, as the system can infer the superclass `pdtb:Causal` from the DimLex-provided senses `pdtb:reason` and `pdtb:result`. The translation pairs of English and German expressions, however, are identical.

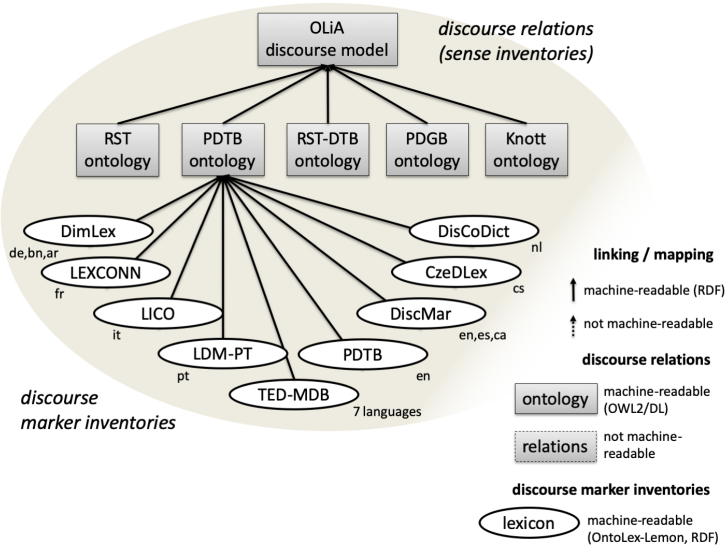


Figure 2 Discourse marker and discourse relation inventories as Linked Data.

Such queries can be further refined if confidence scores or relative sense frequencies are taken into consideration. From the current set of discourse marker lexicons, however, only PDTB and the German DimLex provide such information. For encoding frequency information at a later stage of development, we plan to apply the OntoLex module for Frequency, Attestation and Corpus Information that is currently being developed [17, OntoLex-FrAC].

Overall, we have been able to show that the linked data edition of the discourse marker lexicons and its linking with the OLiA Discourse Extensions provide improved means of querying this data. The example queries have addressed three types of queries:

- discourse marker ↦ PDTB concept ↦ discourse marker (from a given discourse marker, retrieve PDTB-equivalent discourse markers)
- discourse marker ↦ PDTB ontology ↦ discourse marker (use the PDTB ontology for imprecise matches, i.e., more general/more specific senses)
- discourse marker ↦ PDTB ontology ↦ OLiA discourse model ↦ RST ("translate" PDTB relations into another theoretical framework)

To the best of our knowledge, none of these functionalities have been possible before.

A specific benefit of publishing this data as Linked Open Data and under resolvable and persistent URLs is that such queries can be executed independently from any local data base installation. Instead, generic web tools such as the "general purpose SPARQL processor" from `http://sparql.org` can be employed to execute such queries.

6 Summary and Outlook

In this paper we described the conversion of existing discourse marker lexicons into RDF, their linking with the PDTB ontology of the OLiA Discourse Extensions and their publication as Linked Data. This contribution is an important step in formation of a small group of discourse-related resources within the Linguistic Linked Open Data cloud. The general structure and the relation between the resources introduced or described in this paper is illustrated in Fig. 2.

The respective discourse marker lexicons are provided as plain RDF dumps (preserving all information from the original XML file in the `dimlex:` namespace, but lacking PDTB linking) and linked OntoLex-Lemon data sets (preserving only statements that involve OntoLex properties or classes, extended with `ontolex:reference` links to the PDTB ontology. As part of the conversion, we introduced BCP47 language tags to identify the participating languages. It is thus possible to load all discourse marker lexicons into a single RDF graph and query, for example, for correspondences between languages. Moreover, machine-readable language identification and adherence to web standards allows us now to explore synergies with other OntoLex-Lemon datasets, e.g., the ACoLi Dictionary Graph [16], e.g., to enrich conventional bilingual dictionaries with machine-readable sense information for discourse markers (in this regard, the PDTB ontology, and the OLiA discourse model can serve a similar function as WordNet for lexical semantics). Likewise, it becomes possible now to explore conventional dictionaries to bootstrap PDTB-linked discourse marker inventories for other languages.

With this kind of data, machine-readable inventories of discourse markers, discourse relations and corpora (resp., their annotation schemes, as formalized in the OLiA Discourse Extensions), it now becomes possible to integrate them into local applications, general web tools, or perform queries against them, as well as enrich them with further information other Linguistic Linked Open Data sets, e.g., general purpose dictionaries provided in OntoLex-Lemon. As the same time, we would like to emphasize that we see prospective users of this technology not so much among specialists in discourse and semantics, but more among developers of technical solutions for studying discourse as well as NLP specialists and knowledge engineers interested in more advanced levels of linguistic analysis and semantic relations beyond individual sentences. As far as the field of discourse studies is concerned, we consider this implementation to provide a practical benefit, but we also assume that general web technologies, e.g., the RDF data model, the Turtle format, and the SPARQL query language, require an additional layer of abstraction in order to be effective tools in the hands of linguist. Such tools are becoming increasingly available for different aspects of linguistic inquiry (e.g. [5, 22, 29]). For discourse studies, such an infrastructure currently does not exist, nor is the use of RDF technologies particularly established in the field, but it is to be noted that the potential for such an application is enormous, as Linked Data provides natural support for standoff and multi-layer annotations [10], all of these are notorious problems for discourse studies [18], as well as for information integration across heterogeneous and distributed data in general, as demonstrated here for discourse marker inventories. By publishing essential data for this field in accordance with Linked Data principles, our work represents an initial step towards the development of advanced tools and improved information aggregation for applications in discourse parsing and discourse analysis.

The OLiA discourse extensions, including the PDTB ontology are published under `http://purl.org/olia/discourse` and available as a code bundle under CC-BY 3.0 from `https://github.com/acoli-repo/olia/tree/master/owl/experimental/discourse`. The discourse marker inventories and the scripts to produce them are currently available under a Apache 2.0 license from `https://github.com/acoli-repo/rdf4discourse`. The data itself remains under the same license as the original data as described above. Code and data is publicly available from our GitHub repository[13] as Open Source under an Apache 2.0 license.

[13] `https://github.com/acoli-repo/rdf4discourse`

References

1 Lasha Abzianidze, Rik van Noord, Hessel Haagsma, and Johan Bos. The first shared task on discourse representation structure parsing. In *Proc. of the IWCS Shared Task on Semantic Parsing*, 2019.

2 Laura Alonso. *Representing discourse for automatic text summarization via shallow NLP techniques*. PhD thesis, Tesis doctoral. Barcelona: Universitat de Barcelona, 2005.

3 Nicholas Asher, Nicholas Michael Asher, and Alex Lascarides. *Logics of conversation*. Cambridge University Press, 2003.

4 Maja Bärenfänger, Mirco Hilbert, Henning Lobin, and Harald Lüngen. Using owl ontologies in discourse parsing. *OTT'06*, 1:87, 2007.

5 Andrea Bellandi, Emiliano Giovannetti, Silvia Piccini, and Anja Weingart. Developing lexo: a collaborative editor of multilingual lexica and termino-ontological resources in the humanities. In *LOTKS-2017*, 2017.

6 Francis Bond and Kyonghee Paik. A survey of wordnets and their licenses. In *Proceedings of the 6th Global WordNet Conference (GWC 2012)*, pages 64–71, Matsue, 2012.

7 Johan Bos. Open-domain semantic parsing with boxer. In *Proceedings of the 20th nordic conference of computational linguistics (NODALIDA 2015)*, pages 301–304, 2015.

8 Peter Bourgonje, Jet Hoek, Jacqueline Evers-Vermeul, Gisela Redeker, Ted Sanders, and Manfred Stede. Constructing a lexicon of dutch discourse connectives. *Computational Linguistics in the Netherlands Journal*, 8:163–175, 2018.

9 Peter Bourgonje and Manfred Stede. Exploiting a lexical resource for discourse connective disambiguation in german. In *Proc. of the 28th International Conference on Computational Linguistics*, pages 5737–5748, 2020.

10 Aljoscha Burchardt, Sebastian Padó, Dennis Spohr, Anette Frank, and Ulrich Heid. Formalising Multi-layer Corpora in OWL/DL – Lexicon Modelling, Querying and Consistency Control. In *Proc. of the 3rd International Joint Conf on NLP (IJCNLP)*, pages 389–396, Hyderabad, India, 2008.

11 Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. Building a discourse-tagged corpus in the framework of Rhetorical Structure Theory. In Jan van Kuppevelt and Ronnie W. Smith, editors, *Current and New Directions in Discourse and Dialogue*, Text, Speech, and Language Technology; 22, chapter 5. Kluwer, Dordrecht, 2003.

12 C. Chiarcos and M. Sukhareva. OLiA - Ontologies of Linguistic Annotation. *Semantic Web Journal*, 518:379–386, 2015.

13 Christian Chiarcos. Towards interoperable discourse annotation. discourse features in the ontologies of linguistic annotation. In *LREC*, pages 4569–4577. Citeseer, 2014.

14 Christian Chiarcos and Tomaz Erjavec. OWL/DL formalization of the multext-east morphosyntactic specifications. In *LAW-2011*, pages 11–20, Portland, Oregon, USA, June 2011. ACL.

15 Christian Chiarcos, Christian Fäth, and Frank Abromeit. Annotation interoperability for the Post-ISOCat era. In *LREC-2020*, pages 5668–5677, 2020.

16 Christian Chiarcos, Christian Fäth, and Maxim Ionov. The ACoLi dictionary graph. In *LREC-2020*, pages 3281–3290, 2020.

17 Christian Chiarcos, Maxim Ionov, Jesse de Does, Katrien Depuydt, Fahad Khan, Sander Stolk, Thierry Declerck, and John Philip McCrae. Modelling frequency and attestations for ontolex-lemon. In *Globalex-2020*, pages 1–9, 2020.

18 Christian Chiarcos, Julia Ritz, and Manfred Stede. Querying and visualizing coreference annotation in multi-layer corpora. In *DAARC-2011*, pages 80–92, 2011.

19 Philipp Cimiano, Christian Chiarcos, John P McCrae, and Jorge Gracia. *Linguistic Linked Data*. Springer, 2020.

20 Philipp Cimiano, John P. McCrae, and Paul Buitelaar. Lexicon Model for Ontologies. Technical report, W3C Community Report, 10 May 2016, 2016.

21 Debopam Das, Manfred Stede, Soumya Sankar Ghosh, and Lahari Chatterjee. DiMLex-Bangla: A lexicon of Bangla discourse connectives. In *LREC*, pages 1097–1102, Marseille, France, 2020. ELRA.

22 Gimena del Rio Riande and Valeria Vitale. Recogito-in-a-box: From annotation to digital edition. *Modern Languages Open*, 2020.

23 S. Farrar and D.T. Langendoen. A linguistic ontology for the semantic web. *Glot International*, 7(3):97–100, 2003.

24 Anna Feltracco, Elisabetta Jezek, Bernardo Magnini, and Manfred Stede. Lico: A lexicon of italian connectives. *CLiC it*, page 141, 2016.

25 Maria Fuentes Fort. *A flexible multitask summarizer for documents from different media, domain and language*. Universitat Politècnica de Catalunya, 2008.

26 Aldo Gangemi, Valentina Presutti, Diego Reforgiato Recupero, Andrea Giovanni Nuzzolese, Francesco Draicchio, and Misael Mongiovì. Semantic web machine reading with fred. *Semantic Web*, 8(6):873–893, 2017.

27 D. Goecke, H. Lüngen, F. Sasaki, A. Witt, and S. Farrar. GOLD and discourse: Domain-and community-specific extensions. In *E-MELD Workshop*, Cambridge, Massachusetts, July 2005.

28 Jorge Gracia, Besim Kabashi, Ilan Kernerman, Marta Lanau-Coronas, and Dorielle Lonke. Results of the translation inference across dictionaries 2019 shared task. In *TIAD*, pages 1–12, 2019.

29 Maxim Ionov, Florian Stein, Sagar Sehgal, and Christian Chiarcos. cqp4rdf: Towards a suite for rdf-based corpus linguistics. In *ESWC-2020*, pages 115–121. Springer, 2020.

30 Alistair Knott and Robert Dale. Using linguistic phenomena to motivate a set of coherence relations. *Discourse processes*, 18(1):35–62, 1994.

31 Harald Lüngen, Maja Bärenfänger, Mirco Hilbert, Henning Lobin, and Csilla Puskás. Discourse relations and document structure. In *Linguistic modeling of information and markup languages*, pages 97–123. Springer, 2010.

32 William C Mann and Sandra A Thompson. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281, 1988.

33 Amália Mendes, Iria del Rio, Manfred Stede, and Felix Dombek. A lexicon of discourse markers for portuguese–ldm-pt. In *LREC-2018*, pages 4379–4384, 2018.

34 Jiří Mírovský, Pavlína Synková, Magdaléna Rysová, and Lucie Poláková. CzeDLex 0.5, 2017.

35 Umangi Oza, Rashmi Prasad, Sudheer Kolachina, Dipti Misra Sharma, and Aravind Joshi. The Hindi discourse relation bank. In *LAW III*, pages 158–161, 2009.

36 Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. The Penn Discourse TreeBank 2.0. In *LREC-2008*, pages 2961–2968, Marrakech, Morocco, 2008.

37 Charlotte Roze, Laurence Danlos, and Philippe Muller. Lexconn: a french lexicon of discourse connectives. *Discours*, 10, 2012.

38 Stephen Soderland, Oren Etzioni, Daniel S Weld, Kobi Reiter, Michael Skinner, Marcus Sammer, Jeff Bilmes, et al. Panlingual lexical translation via probabilistic inference. *Artificial Intelligence*, 174(9-10):619–637, 2010.

39 Caroline Sporleder and Alex Lascarides. Using automatically labelled examples to classify rhetorical relations: An assessment. *Natural Language Engineering*, 14(3):369, 2008.

40 Manfred Stede, Tatjana Scheffler, and Amália Mendes. Connective-lex: A web-based multilingual lexical resource for connectives. *Discours*, 24, 2019.

41 Manfred Stede and Carla Umbach. Dimlex: A lexicon of discourse markers for text generation and understanding. In *COLING-ACL-1998*, pages 1238–1242, 1998.

42 Florian Wolf and Edward Gibson. Representing Discourse Coherence: A Corpus-Based Study. *Computational Linguistics*, 31(2):249–287, 2005.

43 Deniz Zeyrek, Amalia Mendes, Yulia Grishina, Murathan Kurfali, Samuel Gibbon, and Maciej Ogrodniczuk. Ted multilingual discourse bank (ted-mdb): a parallel corpus annotated in the PDTB style. *LREC-2019*, pages 1–38, 2019.

44 Yuping Zhou and Nianwen Xue. PDTB-style discourse annotation of chinese text. In *ACL-2012*, pages 69–77, 2012.

45 Šárka Zikánová, Jiří Mírovskỳ, and Pavlína Synková. Explicit and implicit discourse relations in the prague discourse treebank. In *TSD-2019*, pages 236–248. Springer, 2019.

Tackling Domain-Specific Winograd Schemas with Knowledge-Based Reasoning and Machine Learning

Suk Joon Hong[1]
School of Mathematics, University of Leeds, UK
InfoMining Co., Seoul, South Korea

Brandon Bennett
School of Computing, University of Leeds, UK

Abstract

The *Winograd Schema Challenge* (WSC) is a commonsense reasoning task that requires background knowledge. In this paper, we contribute to tackling WSC in four ways. Firstly, we suggest a keyword method to define a restricted domain where distinctive high-level semantic patterns can be found. A *thanking domain* was defined by keywords, and the data set in this domain is used in our experiments. Secondly, we develop a high-level knowledge-based reasoning method using semantic roles which is based on the method of Sharma [17]. Thirdly, we propose an ensemble method to combine knowledge-based reasoning and machine learning which shows the best performance in our experiments. As a machine learning method, we used Bidirectional Encoder Representations from Transformers (BERT) [3, 9]. Lastly, in terms of evaluation, we suggest a "robust" accuracy measurement by modifying that of Trichelair et al. [20]. As with their switching method, we evaluate a model by considering its performance on trivial variants of each sentence in the test set.

2012 ACM Subject Classification Computing methodologies → Artificial intelligence

Keywords and phrases Commonsense Reasoning, Winograd Schema Challenge, Knowledge-based Reasoning, Machine Learning, Semantics

Digital Object Identifier 10.4230/OASIcs.LDK.2021.41

Related Version *Full Version*: `https://arxiv.org/abs/2011.12081`

Supplementary Material *Software (Source Code)*: `https://github.com/hsjplus/high-level-kb-reasoning`; archived at `swh:1:dir:bf9138cbf3a41a02809ac1de2dea41d499b9198e`

1 Introduction

The *Winograd Schema Challenge* (WSC) was proposed by Levesque et al. [10] as a means to test whether a machine has human-like intelligence. It is an alternative to the well known *Turing Test* (TT) and has been designed with the motivation of reducing certain problematic aspects that affect the TT. Specifically, while the TT is subjective in nature, the WSC provides a purely objective evaluation; and, whereas passing the TT requires a machine to behave in a deceptive way, the WSC takes the form of a positive demonstration of intelligent capability.

The core problem of the WSC is to resolve the reference of pronouns occurring in natural language sentences. To reduce the possibility that the task can be accomplished by procedures based on superficial or statistical characteristics, rather than "understanding" of the sentence, it is required that the test sentences used in the WSC should be constructed in pairs, which have similar structure and differ only in some key word or phrase, and such that the correct referent of the pronoun is different in the two cases. This sentence pair, together with an

[1] corresponding author

indication of which pronoun is to be resolved and a pair of two possible candidates, is called a *Winograd Schema*. An example of a Winograd Schema from the original WSC273 data set [10] is as follows:

1. The trophy doesn't fit in the brown suitcase because **it** is too *large*.
   - **Candidates for the pronoun:** the trophy / the suitcase, **Answer:** the trophy
2. The trophy doesn't fit in the brown suitcase because **it** is too *small*.
   - **Candidates for the pronoun:** the trophy / the suitcase, **Answer:** the suitcase

Levesque et al. [10] design Winograd schemas to require background knowledge to resolve a pronoun, which can be an evidence of *understanding*. Therefore, they aim to exclude the sentences that can be resolved by a superficial statistical association within a sentence.

In this paper, we used a keyword method to define domains in Winograd schemas. To our knowledge, this is the first work to use keywords for defining domains in WSC and explore high-level patterns in them. To use the domain-specific high-level patterns, we also develop an advanced high-level knowledge-based reasoning method by modifying the method of Sharma [17]. Furthermore, we suggest a simple ensemble method that combines knowledge-based reasoning and machine learning. By the experiments on the domain-specific data set, the ensemble method gives a better performance than each single method. Lastly, we also propose a "robust" accuracy measure that is more objective by improving the switching method of Trichelair et al. [20].

2 Related work

Knowledge-based reasoning and machine learning are the two main approaches to resolve Winograd schemas.

Knowledge-based reasoning

The paper of Levesque et al. [10] is concerned with defining a test for AI rather than proposing how the challenge should be addressed. However, in the paper's conclusion they suggest that the knowledge representation (KR) approach is the most promising. They say: *"While this approach (KR) still faces tremendous scientific hurdles, we believe it remains the most likely path to success. That is, we believe that in order to pass the WSC, a system will need to have commonsense knowledge about space, time, physical reasoning, emotions, social constructs, and a wide variety of other domains."*

KR techniques make use of explicit symbolic representations of information and inference rules. A number of researchers have taken this kind of approach. Bailey et al. [1, p. 18] propose a "correlation calculus" for representing and reasoning with background knowledge principles and use this to derive solutions to certain Winograd schemas. Sharma [17] employs automated extraction of graphical representations of a sentence structure using a semantic parser called K-Parser [18] and implements a WSC resolution procedure based on Answer Set Programming (ASP) [5].

An advantage of KR-based methods is that they provide explanations of how the answers they give are justified by logical prinicples. However, KR-based methods also face huge problems both in automating the conversion from natural language sentences to a formal representation and also in building a knowledge base that covers the *general domain* of knowledge required to address the WSC. Bailey et al. [1] do not give an automatic method to transform a natural language sentence into the form of first-order logic that they use. Though Sharma et al. [19] do use an automated method to extract background knowledge, their method is based on using a search engine, which cannot guarantee acquiring all necessary knowledge.

Table 1 Two Examples from WSC273 with each variant by negation on which Kocijan's BERT was tested.

Type	Sentence	Pred.	Answer
Ori.	Dan had to stop Bill from toying with the injured bird. **He** is very compassionate.	**Dan**	Dan
Neg.	Dan had to stop Bill from toying with the injured bird. **He** is **not** compassionate.	**Dan**	Bill
Ori.	I can't cut that tree down with that axe; **it** is too small.	**The tree**	The axe
Neg.	I can't cut that tree down with that axe; **it** is **not** small.	**The tree**	The tree

Machine learning

Contrary to the expectations expressed by the proposers of the challenge (as cited in the previous section), many researchers have applied Machine Learning (ML) methods to the WSC, and, in terms of accuracy performance, impressive results have been obtained. An early work by Rahman and Ng [13] extracts features of a WSC-like sentence by using background knowledge such as Google search counts and a large corpus, and these features are used to train the SVM ranker that gives the higher rank to the correct candidate.

More recent ML approaches mostly use a neural language model. Trinh and Le [21] introduce an approach to use a neural language model to tackle Winograd schemas. After this, Bidirectional Encoder Representations from Transformers (BERT) [3], which is a state-of-the-art language model, is also used for WSC. Kocijan et al. [9] demonstrate that the BERT fine-tuned with the data set similar to Winograd schemas gives a better performance than the BERT without fine-tuning. In addition, Sakaguchi et al. [16] give the accuracy of around 90% on the original WSC273 by fine-tuning a variant of BERT with the larger data set (WinoGrande) which is also similar to Winograd schemas.

Despite the high accuracy of BERT and other neural language model methods, some limitations have been found. Though many of the original Winograd schemas can be resolved by the language models, Trichelair et al. [20] demonstrate that they often predict wrongly on simple variants of the original sentences. Specifically, when we switch the positions of the candidates, in most cases this means that the answer should also be switched. However, the language model methods frequently give the same prediction for the switched sentence as in the original sentence. We return to this matter of switching in Section 6. Their finding implies that the real understanding of the model cannot be guaranteed by accuracy only. Furthermore, Ettinger [4] also shows that the BERT does not seem to understand negation since BERT's predictions on the masked tokens of the negated sentences are likely to be similar to its predictions on the masked tokens of the non-negated sentences.

The finding of Ettinger [4] is also supported by recent study [11] and the experiments of Kocijan's BERT on some Winograd schema sentences from WSC273 that are negated by us in Table 1. Though the answers should be changed on the negated Winograd schema sentences in this example, the BERT's predictions on them are still same as its predictions on the non-negated sentences.

Table 2 The five major high-level domain-specific reasoning patterns found in the thanking domain.

Type	Sentence
Pattern 1	Candidate1 owes candidate2, and (so) pronoun is doing good
Pattern 2	Candidate1 owes candidate2, and (so) pronoun is receiving good
Pattern 3	Candidate1 does good to candidate2 because pronoun is owing
Pattern 4	Candidate1 gives thanks to candidate2 because pronoun is being owed
Pattern 5	Candidate1 gives thanks to candidate2 because pronoun is owing

3 Semantic Domains and Keywords

Several researchers in natural language processing have suggested that semantic domains can be identified based on the occurrence of key words in text corpora [14, 6]. Assuming that keywords are related to the high-level semantic meaning of a sentence, we used a keyword method in terms of identifying a domain in Winograd Schemas. To our best knowledge, our method is the first work to use keywords regarding a domain in Winograd schemas and examine high-level patterns in a domain. Although defining a domain by keywords has weakness such as word sense ambiguity, it can be beneficial for knowledge-based reasoning which requires relevant knowledge to tackle WSC. A keyword-based domain could target narrower Winograd schema sentences that are related to smaller number of background knowledge principles since they share at least one word. In this sense, building a knowledge base for a keyword-based domain can be less costly.

For the pilot study, we chose a *thanking* domain since the thanking domain has a distinctive semantics. The thanking domain contains the sentences that have a keyword related to the normal sense of thanking. The keywords we used for the thanking domain were "thank" and "grateful". We extracted sentences that include the two keywords from WinoGrande [16] which has approximately 44K Winograd schema sentences since WSC273 contains only 273 sentences. In this extraction, we exclude the sentences including "thanks to" and "thanks in no small part to" though "thank" is within them. The reason for their exclusion is that their semantic meaning is related to causal relations, not thanking.

As a result, the number of the extracted Winograd schema sentences was 171 ($\approx 0.39\%$ of the $44,000$ Winogrande sentences). We believe that the number of them is adequate as it is compatible with the number of the original WSC273's sentences which is 273. These extracted sentences are considered to belong to the thanking domain, and we investigated the high-level reasoning patterns in the thanking domain. As shown in Table 2, the five major high-level domain-specific reasoning patterns were found. As these patterns are from the thanking domain, they are related to the relationships of "owing" and "being owed". It is common that a person who is owing would thank or do good to someone who is owed. It is interesting that around 77% (132/171) of the sentences in the thanking domain follow the only five major high-level patterns. Some of the other minor high-level patterns were also found in the thanking domain.

In addition to the high-level patterns, the Winograd schema sentences in the thanking domain have two other characteristics. The first characteristic is that more than 90% (161/171) of the sentences in the thanking domain have candidates with human names while this proportion is around 50% in WSC273. This finding can be explained by the fact that thanking is done by humans. For the second characteristic, only around 46% (80/171) of the sentences in the thanking domain can be paired while almost all the sentences can be paired in WSC273. This is due to the fact that some of the WinoGrande sentences use keywords such as "thank" for the special words or the alternative words.

4 The advanced high-level knowledge-based reasoning method

Our high-level knowledge-based reasoning method is related to the method of Sharma [17], who identifies and exploits very specific identity implications to resolve pronouns. We use a more general method of abstracting semantic relationships to identify and make use of high-level domain-specific semantic roles, based on the analysis of Winograd schemas given by Bennett [2]. According to this analysis, most Winograd sentences can be represented as having the form:

$$\phi(a, b, p) \equiv ((\alpha(a) \ \wedge \ \beta(b) \ \wedge \rho(a, b)) \ \# \ \pi(p)) \tag{1}$$

where α is the candidate a's property, β is the candidate b's property, ρ refers to a predicate that defines the candidates' relationship, # refers to the relationship between the clause of the sentence that contains candidates and the clause of the sentence that contains the pronoun, and π is the pronoun p's property. In the most common cases the relationship # is "because", but it can also be other connectives such as "and", "but", "since", or sometimes just a full stop between sentences. For instance, consider this sentence from WinoGrande:

Lawrence thanked **Craig** profusely for the assistance ... because only [**he**] helped him.

Here a and b correspond to Lawrence and Craig, and the predicates α and β refer to the roles thanker and being thanked. p corresponds to the pronoun ("he") and the predicate π refers to the role of helper. ρ can refer to (a) giving thanks to (b) and # can be "because". While this type of formula can be used for particular examples of Winograd schemas, we also used the formula to represent higher-level general principles that can potentially explain a large class of specific cases.

4.1 Building a domain-specific knowledge base

Our knowledge base is composed of two types of rules and one type of facts – rules to derive semantic roles, rules to define relationships regarding the semantic roles and high-level background knowledge principles.

Rules to derive semantic roles

We defined rules to derive semantic roles specific to the thanking domain. These semantic roles are high-level representations related to the candidates and the pronoun, and they are also grounds to derive the relationships regarding them. In the thanking domain, six major domain-specific semantic roles were found – thanker, being thanked, giver, given, helper and being helped. In the current work, we assume that each person has a role in relation to the situation being described, and we formulate rules to derive and reason about these roles. (Potentially, someone could have different roles with respect to different aspects of the situation, which would require elaboration of our framework.)

Table 3 The major rules to define the relationships between the semantic roles of the candidates.

Semantic relationship	Causal relation	Semantic role	
		X	Y
X owes Y	No	being helped	helper
X owes Y	No	given	giver
X does good to Y	Yes	helper	being helped
X does good to Y	Yes	giver	given
X gives thanks to Y	Yes	thanker	being thanked

Our rules are implemented in ASP by using K-Parser's graphical representations, and they are manually defined from the sentences in the thanking domain. For example, a simple rule for thanker can be defined as:

```
has_s(X, semantic_role, thanker) :-
        has_s(Thank,agent,X),
        has_s(Thank,instance_of,thank).
```

In order to make more generalisable rules, the following four measures were taken. The first measure is to derive the semantic role of a candidate if that of the other candidate is known (e.g. if "give" is the semantic role of a candidate, then that of the other candidate would be "given"). The second measure is for the case when no semantic roles of the candidates are known. For instance, if candidate1 is an agent of the verb to which candidate2 is a recipient, candidate1's semantic role is derived to be "giver". The third measure is to use synonyms that are manually defined in the thanking domain. The fourth measure is to use an external sentiment lexicon dictionary [8] to derive the semantic roles of "good" and "bad".

Rules to define relationships regarding the semantic roles

The domain-specific semantic roles are used to derive their relationships for the high-level representations of Winograd schema sentences. We defined the rules for the relationships using the semantic roles in the following three aspects: relationships between the semantic roles of the candidates, relations between the clause containing the candidates and the clause containing the pronoun, and property of the pronoun.

1. **Relationships between the candidates' semantic roles.**
   As the five high-level patterns in Table 2 show, the two candidates in a Winograd schema are found to have domain-specific relationships in the thanking domain. The main relationships between them are "owes", "does good to" and "gives thanks to". In order to derive the relationships between the semantic roles of the candidates, we defined the rules by using their semantic roles and the existence of causal relation. Table 3 shows the five rules to derive the relationships between the candidates.

   For instance, the second rule in Table 3 means that if the semantic role of X is "given", that of Y is "giver", and there is no causal relation then X owes Y. It is written in ASP as:

   ```
   has_s(X, owes, Y) :-
       has_s(X,semantic_role,given),
       has_s(Y,semantic_role,giver),
       not has_s(_,relation,causer).
   ```

2. **The relationship between first and second clauses of the sentence.**
As represented in Formula (1), the structure of a Winograd schema involves a relationship between the first clause of the sentence containing the candidates and the second clause containing the pronoun ("#"). In most cases we assume that there is some kind of implication from the first clause to the second clause, corresponding to some reasoning principle. However, if the sentence is of the form "P *because* Q", then the implication will go from the second clause to the first(Q to P). In this second case, K-Parser generates a `caused_by` relationship. Hence, we have a rule that when this relation is present, the agent of the second clause (i.e. the pronoun reference) has a causal role in the first clause of the sentence (i.e. corresponds to the candidate who is the agent in the first part). This rule can be defined in ASP as follows:

```
has_s(P, relation, causer) :-
    pronoun(P),
    is_candidate(A),
    has_s(Verb1,caused_by,Verb2),
    1 {has_s(Verb1,agent,A);
    has_s(Verb1,recipient,A)},
    has_s(Verb2,agent,P).
```

3. **Property of the pronoun.**
The semantic role of the pronoun can be the property of the pronoun ("$\pi(p)$") in Formula (1), but there can be a higher-level semantic role. For this reason, we defined rules to derive the high-level semantic role from the low-level semantic role. These rules are based on the fact that a low-level semantic role can be *a subset of* a high-level semantic role in the thanking domain. For instance, the semantic role of "helper" can be a subset of that of "doing good". We implemented these rules in ASP, and the following rule is one of them:

```
has_s(X, semantic_role, doing_good) :-
    has_s(X, semantic_role, helper).
```

High-level background knowledge principles

In our knowledge base, we also defined high-level domain-specific background knowledge principles as well as the two types of the rules above. The high-level background knowledge principles are used for the reasoning in comparison with the high-level representation of a sentence that is derived by the rules in the knowledge base. We followed the style of Sharma [17]'s background knowledge principles as a foundation, but different from Sharma [17], our background knowledge principles are based on the semantic roles' relationships derived by our knowledge base.

4.2 Transforming a Winograd schema sentence into a high-level representation

We used K-Parser to transform the Winograd schema sentences in the thanking domain into the graphical representations as Sharma [17] does. By using the rules to derive semantic roles and to derive relationships between the semantic roles, we transformed the graphical representations into high-level representations. The following is an example of the transformations from WinoGrande:

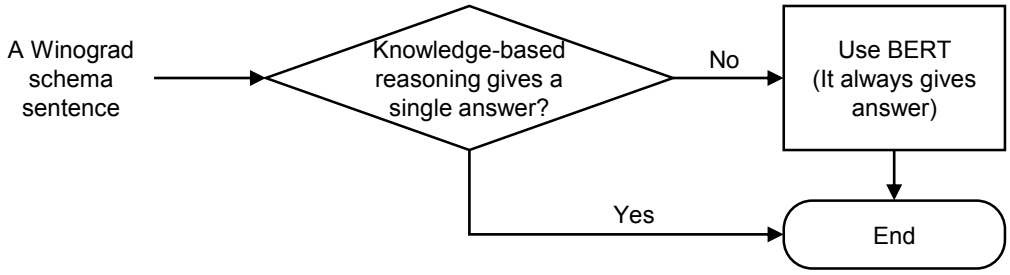


Figure 1 Our algorithmic flow of combining the knowledge-based reasoning method and the machine learning method.

Kayla cooked sticky white rice for **Jennifer**, and [**she**] was thanked for making such delicate rice.

- **The semantic roles:**
  1. Kayla: giver
  2. Jennifer: given
  3. she: being thanked
- **The relationships regarding the semantic roles:**
  1. Jennifer **owes** Kayla
  2. no causal relation
  3. she is receiving good

4.3 Reasoning to derive the answer

We used the reasoning rules of Sharma [17] with small modifications to resolve the Winograd schemas in the thanking domain. The goal of the modifications was to use the derived semantic roles for the reasoning.

In the reasoning process, each Winograd schema sentence is compared with each background knowledge principle. As a result, the answer for each sentence can be a single answer, "no answer" and multiple answers. If multiple answers have the same answers, this case is considered as a single answer.

As an example of the reasoning method, suppose a background knowledge principle is given in Sharma's form [17] as:

IF someone **owes** a person $p1$, and (consequently) a person $p2$ is receiving good **THEN** $p1$ is same as $p2$. (There is an assumption that owing occurs before receiving good.)

This background knowledge principle corresponds to the derived relationships regarding the semantic roles in the previous subsection. By applying the reasoning rules, $p1$ and $p2$ in the background knowledge principle correspond to "Kayla" and "she" in the sentence. Thus, the answer "she" = "Kayla" can be derived.

5 The simple ensemble method

We combined our advanced high-level knowledge-based reasoning method with Kocijan's BERT [9]. The aim of our ensemble method is to mitigate each method's weakness, and recent research [7] also suggests that machine learning and knowledge-based reasoning can complement each other. The weakness of the advanced high-level knowledge-based reasoning

method is that if there are no rules that can be applied in the knowledge base, no answer can be derived. With respect to weakness of language models such as BERT, their predictions are vulnerable to the small changes since it is not based on a logical relationship [20, 4].

As shown in Figure 1, we implemented a simple but effective ensemble method. If the knowledge-based reasoning method gives a single answer, the final answer will be this answer. On the other hand, if the prediction of the knowledge-based reasoning method is multiple answers or no answer, we use the BERT's prediction for the final answer. With these two conditions, the weakness of each method can be reduced.

6 "Robust" accuracy

As mentioned in Section 2, machine learning methods can give the incorrect answer on trivial variants of sentences obtained by switching the candidates [20]. This reveals an apparent weakness in these methods and a limitation in the simple evaluation of accuracy. Accuracy measurement is already quite tolerant because, since the number of the candidates are only two, the chance of predicting correctly without understanding is 50%. This is a further motivation for having a stricter form of accuracy measurement. We propose a "robust" accuracy measurement based on a generalisation of Trichelair et al. [20]. In addition to the switching, we add three more variants of each sentence by replacing the name of each candidate with the random name with the same gender if the candidates are both names. This basic method of replacing names should not affect the fundamental meaning of a sentence, and thus a model's incorrect predictions on the sentences where only the names are replaced can reveal its obvious lack of understanding. The following is an original sentence from WinoGrande in the thanking domain and its variants to measure the robust accuracy:

- **Original sentence**: **Kayla** cooked sticky white rice for **Jennifer**, and [she] was thanked for making such delicate rice.
- **The nouns switched**: **Jennifer** cooked sticky white rice for **Kayla**, and [she] was thanked for making such delicate rice.
- **The nouns replaced 1**: **Tanya** cooked sticky white rice for **Kayla**, and [she] was thanked for making such delicate rice.
- **The nouns replaced 2**: **Erin** cooked sticky white rice for **Tanya**, and [she] was thanked for making such delicate rice.
- **The nouns replaced 3**: **Lindsey** cooked sticky white rice for **Christine**, and [she] was thanked for making such delicate rice.

Only when a model predicts correctly on all of the original Winograd schema sentence and the four variants including the switched one, that prediction is considered to be "robustly" accurate. While the probability of predicting correctly on both switched and non-switched sentences out of luck is $0.5 \times 0.5 = 0.25$, the probability can go down to $(0.5)^5 \approx 0.03$ in the robust accuracy. In this sense, our robust accuracy is more objective on evaluating a model's performance. The limitation of the robust accuracy is that the candidates should be human names to make variants. In the case of no human names for the candidates, we only used the switching method to make a variant. This kind of exception is not common in the thanking domain since more than 90% of the sentences have the candidates with human names.

7 Evaluation

Our evaluation compares the performance of the following methods: GPT-2 [12], BERT-large [3], Kocijan's BERT-large [9], Kocijan's BERT-large further fine-tuned with the domain train set, our advanced high-level knowledge-based reasoning method and our ensemble method. When GPT-2 was used for resolving Winograd Schemas, partial scoring [21] was used to calculate the sentence probability of each candidate replacing the pronoun. Kocijan's BERT we used is their best performing model ("BERT_WIKI_WSCR") [9] which was fine-tuned with the WSC-like sentences[13]. We implemented Kocijan's BERT for our experiments by using the model and the code in their repository[2].

The six different methods were evaluated on the 80 paired Winograd schema sentences in the thanking domain, and the 91 non-paired sentences were used for validation. For the evaluation metrics, we used accuracy and our stricter "robust" accuracy measure.

We did two experiments with the paired sentences in the thanking domain. In the first experiment, each pair was *split* , so that one of the pair was put into the train set and the other into the test set. By its definition, 50% of the paired sentences were used for the train set, and the others were used for the test set. In the second experiment, on the other hand, each pair was put *together* either both in the train set or both in the test set in a random manner. Considering the small number of the data set and the balance with the first experiment, the second experiment also took the $50 : 50$ split between the train set and the test set.

7.1 Results

Tables 4 and 5 show the results of the two experiments respectively. Some same patterns were found in both experiments. The accuracies and the robust accuracies of our ensemble model are better than those of the other methods. Also, the models that contain a language model were found to have the lower robust accuracies than the accuracies. It demonstrates that language models, as machine learning methods, can be weak to minor changes.

Different patterns were also found between the two experiments. The accuracy of the knowledge-based reasoning method in the first experiment was higher than that in the second experiment by a large margin. It implies that the close similarity between the train set and the test set is advantageous for the knowledge-based reasoning method since the rules defined by the train set are expected to be used for the test set.

On the other hand, Kocijan's BERT-large further fine-tuned with the domain train set [9] gave the opposite results since the better accuracy was found in the second experiment, not in the first experiment. This result can be explained by the characteristics of Winograd schemas. While similar sentences have different answers in a Winograd schema, language models such as BERT are likely to give the same answer with that of the similar sentence, which leads to the wrong predictions in the first experiment. This result is compatible with the finding of Kocijan et al. [9] that training with the paired sentences shows a better performance than training with the non-paired sentences.

It is interesting that GPT-2 [12] and BERT-large [3] show the large gaps equal to or over 20% between accuracy and the "robust" accuracy in both experiments when they are not fine-tuned with WSC-like sentences. In contrast, the Kocijan's BERT-large models where fine-tuning was applied show the smaller gaps below 10% between accuracy and the "robust"

[2] `https://github.com/vid-koci/bert-commonsense`

Table 4 The results of the first experiment. These methods were tested on the same test set in the thanking domain with each pair split (between the train set and the test set).

Model	Accuracy	"Robust" accuracy
GPT-2 (no further fine-tuning) [12]	50.0% (20/40)	20.0% (8/40)
BERT-large (no further fine-tuning) [3]	57.5% (23/40)	37.5% (15/40)
Kocijan's BERT-large fine-tuned with the WSC-like data set [9]	70.0% (28/40)	62.5% (25/40)
Kocijan's BERT-large further fine-tuned with the domain train set	47.5% (19/40)	42.5% (17/40)
Our knowledge-based reasoning method	72.5% (29/40)	72.5% (29/40)
Our knowledge-based reasoning method + Kocijan's BERT-large [9] fine-tuned with the WSC-like data set[13]	**90.0**% (36/40)	**85.0**% (34/40)

accuracy in both experiments. This finding implies that the fine-tuning method applied to Kocijan's BERT-large can make language models more robust in terms of tackling Winograd schemas.

8 Conclusion

This paper demonstrates that combining both the high-level knowledge-based reasoning method and the BERT can give a better performance in the thanking domain.

In this paper, we also used the keywords method to identify a domain, and this method can be applied to specify other domains. We showed that high-level patterns were found in the domain defined by the keywords. As only one domain – the thanking domain – was tackled, future work needs to be done with more domains in Winograd schemas. Though the number of the thanking domain is 171 (around 0.39% of the number of the WinoGrande) as a pilot study, some other domains could be larger than the thanking domain. For instance, the domain that can be defined by the keywords "love" and "hate" has $1,351$ (around 3%) and 612 (around 1%) sentences respectively. If these were genuinely separate domains and the correct resolution of each schema were based on principles in the domain corresponding to the key words it contains, this would imply that tackling around 100 domains could cover almost all domains in Winograd schemas.

By modifying the method of Sharma [17] and focusing on the domain-specific semantic roles, we were able to develop a knowledge-based reasoning method that can use domain-specific high-level patterns. Though our knowledge-based method uses background knowledge principles that are built manually, we believe that our principles are more accurate than the kinds of semantic feature that could be reliably extracted from a large corpus or by using a search engine. This is because the simple statistical method used for automatically extracting

Table 5 The results of the second experiment. These methods were tested on the same test set in the thanking domain with pairs kept together (either both in the train set or both in the test set).

Model	Accuracy	"Robust" accuracy
GPT-2 (no further fine-tuning) [12]	57.5% (23/40)	15.0% (6/40)
BERT-large (no further fine-tuning)[3]	57.5% (23/40)	35.0% (14/40)
Kocijan's BERT-large [9] fine-tuned with the WSC-like data set[13]	77.5% (31/40)	70.0% (28/40)
Kocijan's BERT-large further fine-tuned with the domain train set	75.0% (30/40)	70.0% (28/40)
Our knowledge-based reasoning method	37.5% (15/40)	37.5% (15/40)
Our knowledge-based reasoning method + Kocijan's BERT-large [9] fine-tuned with the WSC-like data set[13]	**80.0**% (32/40)	**72.5**% (29/40)

knowledge is vulnerable to data bias or special usage of words in idioms (e.g. "thanks to" referring to causal relations that do not involve thanking in the normal sense of this concept). In addition, our knowledge-based method can also be used in other natural language tasks such as Choice Of Plausible Alternaties (COPA) [15]. But K-Parser used in our approach still needs to be improved as manual corrections were needed in some cases.

We also proposed the robust accuracy by improving the method of Trichelair et al. [20]. The decreased robust accuracies of language models such as BERT and GPT-2 reveal that their accuracy may not entail their real understanding.

Code repository

The code for the advanced high-level knowledge-based reasoning method (described in Section 4) can be accessed from the following repository: `https://github.com/hsjplus/high-level-kb-reasoning`

References

1 Daniel Bailey, Amelia Harrison, Yuliya Lierler, Vladimir Lifschitz, and Julian Michael. The winograd schema challenge and reasoning about correlation. In *2015 AAAI Spring Symposium Series*, USA, 2015.

2 Brandon Bennett. Logical analysis of winograd schemas. *Unpublished*, 2020.

3 Jacob Devlin, Ming W. Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv [cs.CL]*, 2018. `arXiv:1810.04805`.

4 Allyson Ettinger. What bert is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48, 2020.

5 Michael Gelfond and Vladimir Lifschitz. The stable model semantics for logic programming. In *Proceedings of International Logic Programming Conference and Symposium*, pages 1070–1080, 1988.

6 Alfio Gliozzo and Carlo Strapparava. *Semantic Domains in Computational Linguistics*. Springer Berlin Heidelberg, 2009.

7 Pascal Hitzler, Federico Bianchi, Monireh Ebrahimi, and Md Kamruzzaman Sarker. Neural-symbolic integration and the semantic web. *Semantic Web*, 11(1):3–11, 2020.

8 Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD2004)*, 2004.

9 Vid Kocijan, Ana M. Cretu, Oana M. Camburu, Yordan Yordanov, and Thomas Lukasiewicz. A surprisingly robust trick for winograd schema challenge. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4837—4842, 2019.

10 Hector Levesque, Ernest Davis, and Leora Morgenstern. The winograd schema challenge. In *The 13th International Conference on Principles of Knowledge Representation and Reasoning*, Italy, June 2012.

11 Kaixin Ma, Jonathan Francis, Quanyang Lu, Eric Nyberg, and Alessandro Oltramari. Towards generalizable neuro-symbolic systems for commonsense question answering. In *Proceedings of the First Workshop on Commonsense Inference in Natural Language Processing*, pages 22–32. Association for Computational Linguistics, 2019.

12 Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners, 2019.

13 Altaf Rahman and Vincent Ng. Resolving complex cases of definite pronouns: The winograd schema challenge. In *EMNLP-CoNLL*, 2012.

14 Paul Rayson. From key words to key semantic domains. *International Journal of Corpus Linguistics*, 13(4):519–549, 2008.

15 Melissa Roemmele, Cosmin A. Bejan, and Andrew S. Gordon. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *AAAI Spring Symposium on Logical Formalizations of Commonsense Reasoning*, USA, March 2011.

16 Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. In *AAAI-20*, 2020.

17 Arpit Sharma. Using answer set programming for commonsense reasoning in the winograd schema challenge. *arXiv [cs.AI]*, 2019. `arXiv:1907.11112`.

18 Arpit Sharma, Nguyen H. Vo, Somak Aditya, and Chitta Baral. Identifying various kinds of event mentions in k-parser output. In *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 82–88. Association for Computational Linguistics, 2015.

19 Arpit Sharma, Nguyen H. Vo, Somak Aditya, and Chitta Baral. Towards addressing the winograd schema challenge - building and using a semantic parser and a knowledge hunting module. In *IJCAI 2015*, pages 1319–1325, 2015.

20 Paul Trichelair, Ali Emami, Adam Trischler, Kaheer Suleman, and Jackie C. K. Cheung. How reasonable are common-sense reasoning tasks: A case-study on the winograd schema challenge and swag. *arXiv [cs.LG]*, 2018. `arXiv:1811.01778`.

21 Trieu H. Trinh and Quoc V. Le. A simple method for commonsense reasoning. *arXiv [cs.AI]*, 2018. `arXiv:1806.02847`.