Entity Set Expansion from the Web via ASP

Weronika T. Adrian[1], Marco Manna[2], Nicola Leone[3], Giovanni Amendola[4], and Marek Adrian[5]

1 **Universty of Calabria, Arcavacata di Rende (CS), Italy and AGH University of Science and Technology, Kraków, Poland**
w.adrian@mat.unical.it
2 **Universty of Calabria, Arcavacata di Rende (CS), Italy**
manna@mat.unical.it
3 **Universty of Calabria, Arcavacata di Rende (CS), Italy**
leone@mat.unical.it
4 **Universty of Calabria, Arcavacata di Rende (CS), Italy**
amendola@mat.unical.it
2 **Universty of Calabria, Arcavacata di Rende (CS), Italy**
m.adrian@mat.unical.it

Abstract

Knowledge on the Web in a large part is stored in various *semantic resources* that formalize, represent and organize it differently. Combining information from several sources can improve results of tasks such as recognizing similarities among objects. In this paper, we propose a logic-based method for the problem of entity set expansion (ESE), i.e. extending a list of named entities given a set of seeds. This problem has relevant applications in the Information Extraction domain, specifically in automatic lexicon generation for dictionary-based annotating tools. Contrary to typical approaches in natural languages processing, based on co-occurrence statistics of words, we determine the common category of the seeds by analyzing the semantic relations of the objects the words represent. To do it, we integrate information from selected Web resources. We introduce a notion of an *entity network* that uniformly represents the combined knowledge and allow to reason over it. We show how to use the network to disambiguate word senses by relying on a concept of *optimal common ancestor* and how to discover similarities between two entities. Finally, we show how to expand a set of entities, by using answer set programming with external predicates.

1998 ACM Subject Classification D.1.6 Logic Programming, H.3.3 Information Search and Retrieval, H.3.5 Online Information Services, I.2.4 Knowledge Representation Formalisms and Methods, I.2.7 Natural Language Processing

Keywords and phrases answer set programming, entity set expansion, information extraction, natural language processing, word sense disambiguation

Digital Object Identifier 10.4230/OASIcs.ICLP.2017.1

1 Introduction

The problem we study in this paper goes under the name of *entity set expansion*. Informally, given a set of words called *seeds*, the goal is to extend the original set with new words of the same "sort". For example, starting from *Rome* and *Budapest*, one could expand these seeds with *Amsterdam*, *Athens*, *Berlin*, ..., *Warsaw*, and *Zagreb*, which are also capital cities of European Union member states. But is this the most appropriate way? In fact, an alternative expansion could be made by *Amsterdam*, *Berlin*, *Dublin*, ..., *Paris*, and *Prague*, which are

also Europe's capitals situated on rivers. Moreover, *Rome* is not only a 'capital', but also a 'drama television series', a 'female deity', and many other things, while *Budapest* is also a 'film series' and a 'rock band', apart from being a 'capital' too. Hence, which is the "best" common sort putting together the original words? Are they 'capitals' or 'films'?

The problem of entity set expansion has been widely studied in the NLP community. Several approaches have been proposed, including *bootstrapping algorithms* [10, 13] that starting from a set of seed words, discover patterns in which they appear in a given corpus, then using those patterns find more examples and repeat the process until an end condition is met. The patterns are usually lexico-syntactic, but recently more advanced ways of characterizing the words in a category to be expanded have also been proposed, e.g. *word embeddings* [3, 6]. As far as the corpus is concerned, the great potential of the Web has been recognized and used to extend the set of seeds [4, 11, 9]. Nevertheless, there are several problems with existing approaches. First, the inherent limitation of statistical methods when analyzing the words, is that they do not take into consideration possible different senses of the same word, domain-specific exceptions etc., so methods that work well for generating general lexicons may fail for domains-specific dictionaries, when the meaning of words do not always agree with statistics [5]. Moreover, the intended categories is usually as simple as a 'person' or a 'city'. We would like to go a step further and be able to discover more descriptive categories, by including the properties of the objects represented with the seeds.

To this end, we propose to use knowledge available on the Web, specifically, stored in selected *semantic resources* that represent semantics of objects, their categorization and relations with other objects. We want to use these resources to disambiguate word meanings and discover commonalities among objects represented with them. Once the common category is singled out, we want to utilize the Web-harvested knowledge, specifically stored in the hypernym database built automatically using Hearst-like patterns. This way, our approach combines structural knowledge from the semantic resources for analyzing and understanding objects, and Web-harvested knowledge to extend the set. We propose a model of an *entity network* that will allow to integrate information from several sources and reason over it. We also propose an implementation in answer set programming with external predicates to query semantic resources.

2 Semantic resources and entity networks

Currently, more and more machine-readable knowledge is available on the Web in a form of *semantic resources*, such as WordNet [7], Wikidata (`http://wikidata.org`), BabelNet [8] and WebIsADatabase[12]. These knowledge bases formalize and organize human knowledge about the world in different scope and manners, focus on various dimensions and areas.

To integrate knowledge from such resources, we propose a model that can uniformly represent information acquired from them. The basic notions we will use are *(semantic) entities* and an *(entity) network*. An entity is a pair $\varepsilon = \langle id(\varepsilon), names(\varepsilon)\rangle$, where $id(\varepsilon)$ is the identifier of ε, and $names(\varepsilon)$ is a set of (human readable) terms describing ε. From a syntactic viewpoint, $id(\varepsilon)$ is a set of strings of the form $src : code$ where src identifies the semantic resource where ε is classified, and $code$ is the local identifier within source src, while $names(\varepsilon)$ is a set of strings. For example, $\varepsilon = \langle\{$`wn:08864547`, `wd:Q40`, `bn:00007266n`$\}, \{$`Austria`, `Republic of Austria`$\}\rangle$ is an entity representing the object in real world, the Republic of Austria, referred to in WordNet (abbreviation `wn` with identifier `08864547`), Wikidata (abbreviated `wd` with item identifier `Q40`), and BabelNet (with synset identifier `bn:00007266n`).

From a semantic point of view, entities may refer to three different kinds of objects. Namely, they can either point to (i) individuals, called hereafter *instances*, such as in the previous example, where the entity denotes a particular country, or (ii) concepts that generalize a *class* of objects e.g., $\varepsilon = \langle\{$ `wn:08562388`, `wd:Q6256`, `bn:00023235n` $\}, \{$`country`$\}\ \rangle$ or to (iii) *(semantic) relations* that hold between two objects e.g., $\varepsilon = \langle\{$`wd:P31`$\}, \{$`instance of, is a, ...`$\}\ \rangle$ or $\varepsilon = \langle\{$ `wd:P131`$\}, \{$`is located in, ...`$\}\rangle$ etc. For convenience, we group the entities representing instances and classes into one group, so-called *(knowledge) units*.

An *(entity) network* is a four-tuple $\mathcal{N} = \langle Uni, Rel, Con, type\rangle$ where: (i) Uni is a set of knowledge units, both classes and instances; (ii) Rel is a set of semantic relations; (iii) $Con \subseteq Uni \times Uni$ is a set of ordered pairs denoting that two units are connected via some (one or more) semantic relations; and (iv) $type : Con \to (2^{Rel} \setminus \emptyset)$ is a function that assigns to each connection a set of semantic relations.

To construct an entity network, one may start from either a set of words (i.e. raw strings) or a set of units. To this end, we use Answer Set Programming (ASP) [1] enriched with *external predicates* [2]. External predicates refer to functions (implemented separately) that encapsulate requests to semantic resources and acquire responses. It is easy to extend the current implementation with a new semantic resource: one needs only to add a new rule with an external predicate – a new (typically very simple) function, compatible with the resource's API. In fact, all the rules that query external sources establish new connections and are of the general form: *newCon*(*InputUnit*, *OutputUnit* [, *optionalArg*]*) `:-` *unitID*(*InputUnit*), &*externalPredicate*(*InputUnit*; *OutputUnit*) [, *optionalRestriction*]*.

For example, given a set of seed words, each encoded with a logical fact `seed(`*SeedWord*`)`, we use the following rules in ASP to establish connections `senseOf` from a set of seed words W to the first node of the network representing the meanings of words:

```
senseOf(SeedWord, SenseID) :- seed(SeedWord), &babelnetSense(SeedWord; SenseID).
```

Once we have the first units in the entity network, we can further expand the network with relations of the represented objects, such as hypernymy:

```
bnISA(ID, PID, PLv) :- babelnetID(ID,Lv), &babelnetISA(ID; PID),
                       babelnetDepth(BabelNetMax), Lv<BabelNetMax, PLv = Lv +1.
```

In this rule, the external predicate `&babelnetISA(`*Input; Output*`)` query BabelNet for hypernyms (superclasses) of the given input, and the optional restrictions set the limits on the number of applications of the rule.

3 Entity set expansion

Given the set W of seeds, we solve ESE by performing three major steps described next.

First, we need to understand the objects represented by the seed words. To this end, we construct a network $\mathcal{N}_1$ from W and expand the hypernymy relations via ASP as described above. From WordNet we acquire the taxonomy up to the most general concept: "entity". From other sources, in which the taxonomy is not guaranteed to be acyclic, we get the hypernyms only up to some fixed level. The output of the expansion is a directed acyclic graph, in which we determine the "correct" meanings of the seed words by identifying the "optimal common ancestors" for W. Basically, we identify via ASP program with weak constraints minimum spanning subtrees in the graph, containing one meaning for each word and one common ancestor. The output of this step is a set of units U.

Once we know the single optimal combination of word senses, we proceed to the phase of *category recognition*. In this step, we create a network $\mathcal{N}_2$ starting from the above set of U. First, we determine the common supertypes by asking the semantic resources for hypernyms up to a given limit. Then, we expand the other semantic relations that connect U to other objects. For each shared relation we obtain a set of units that are the *image* of the relation w.r.t. the seed units. If the set is a singleton, it means that the seed units are connected via the relation to the same unit. If it is not the case, then we treat the image set as the new set of seeds, for which we repeat the process of finding a common supertype and analyzing common relations (the iteration limit can be set). The output of this step is a sub-network $\mathcal{N}_3$ that describes the common properties and will be used as "verifier" in the next step.

Finally, to discover new objects of the target category, we query the WebIsADatabase for instances of the common ancestors of the seeds, setting a threshold to filter out noisly results. The obtained set of new candidate instances is then evaluated against the properties discovered earlier. We check if they are hyponyms of one of the desired common ancestors, and if they share the relations discovered for the seed set.

4 Conclusion

The problem of entity set expansion is not a new topic. With our approach, we address the old problem in a moderm semantic way. Instead of relying strictly on lexical level, we utilize the online *semantic resources*, that were not available before, to build a better representation, based on semantic relations. Our approach allows to leverage existing resources, and we believe that with the theoretical foundations and efficient ASP-based implementation of prototypes, that we already have, we can build, with further engineering effort, an integrated, configurable system.

References

1 Gerhard Brewka, Thomas Eiter, and Miroslaw Truszczynski. Answer set programming at a glance. *Communications of the ACM*, 54(12):92–103, 2011.

2 Francesco Calimeri, Davide Fuscà, Simona Perri, and Jessica Zangari. I-DLV: the new intelligent grounder of DLV. *Intelligenza Artificiale*, 11(1):5–20, 2017.

3 José Camacho-Collados, Mohammad Taher Pilehvar, and Roberto Navigli. A unified multilingual semantic representation of concepts. In *Proc. of ACL'15*, pages 741–751, 2015.

4 Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. Unsupervised named-entity extraction from the web: An experimental study. *Artificial Intelligence*, 165(1):91–134, 2005.

5 Ruihong Huang and Ellen Riloff. Inducing domain-specific semantic class taggers from (almost) nothing. In *Proc. of ACL 2010*, pages 275–285, 2010.

6 Ignacio Iacobacci, Mohammad T. Pilehvar, and Roberto Navigli. Sensembed: Learning sense embeddings for word and relational similarity. In *Proc. of ACL 2015*, pages 95–105, 2015.

7 George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

8 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.

9 Patrick Pantel, Eric Crestan, Arkady Borkovsky, Ana-Maria Popescu, and Vishnu Vyas. Web-scale distributional similarity and entity set expansion. In *Proc. of EMNLP 2009*, pages 938–947, 2009.

10 Ellen Riloff and Rosie Jones. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proc. of AAAI '99 and IAAI '99*, pages 474–479, 1999.

11 Luís Sarmento, Valentin Jijkoun, Maarten de Rijke, and Eugenio Oliveira. "more like these": growing entity classes from seeds. In *Proc. of CIKM'07*, pages 959–962, 2007.

12 Julian Seitner, Christian Bizer, Kai Eckert, Stefano Faralli, Robert Meusel, Heiko Paulheim, and Simone Paolo Ponzetto. A large database of hypernymy relations extracted from the web. In *Proc. of LREC'16*, 2016.

13 Michael Thelen and Ellen Riloff. A bootstrapping method for learning semantic lexicons using extraction pattern contexts. In *Proc. of EMNLP '02*, pages 214–221, 2002.