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.

$$y^* = argmax_{Y \in Y_H} S(H,Y) \tag{7}$$

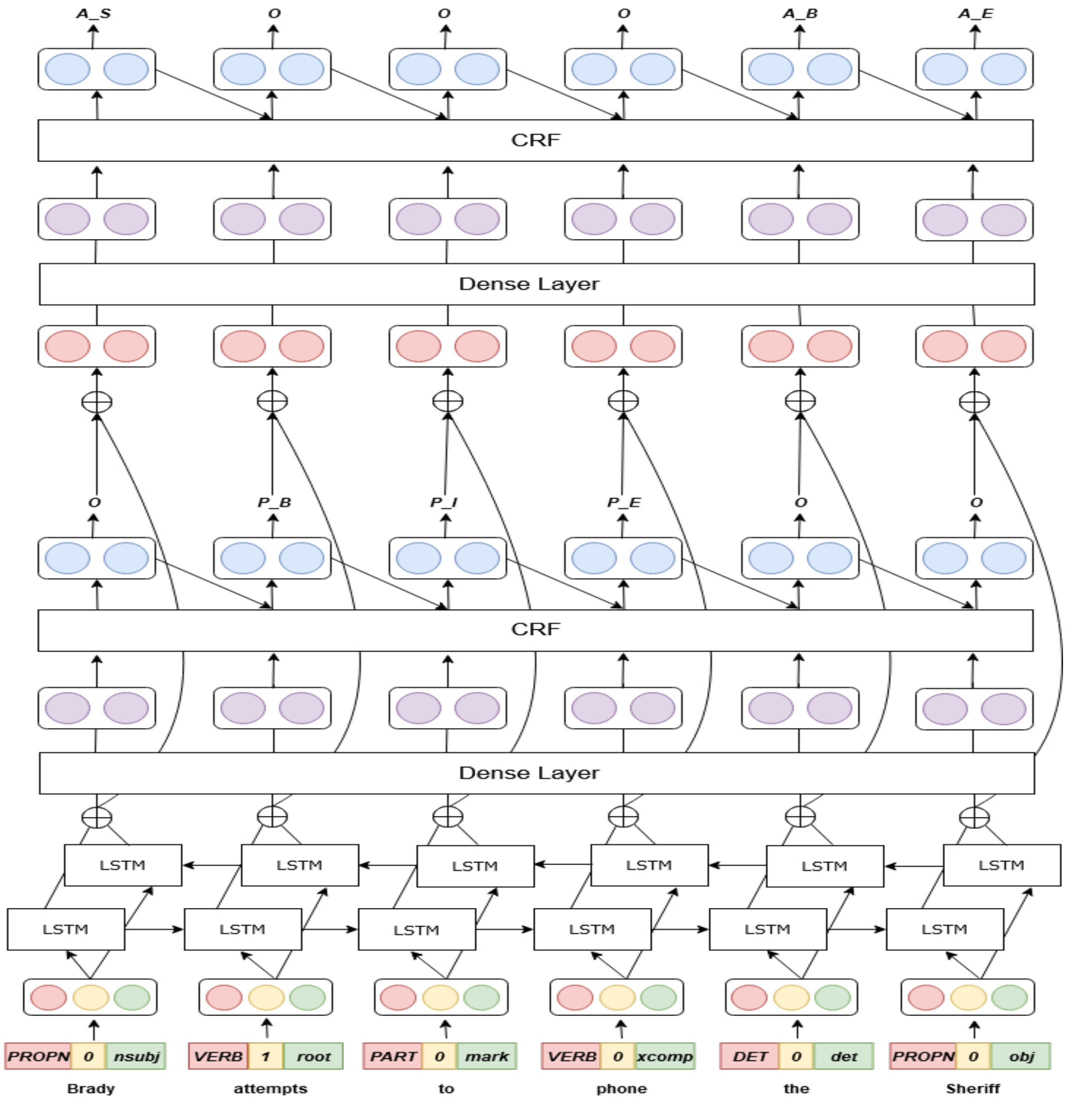

Figure 1 Architecture of the predicate-argument extractor.

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

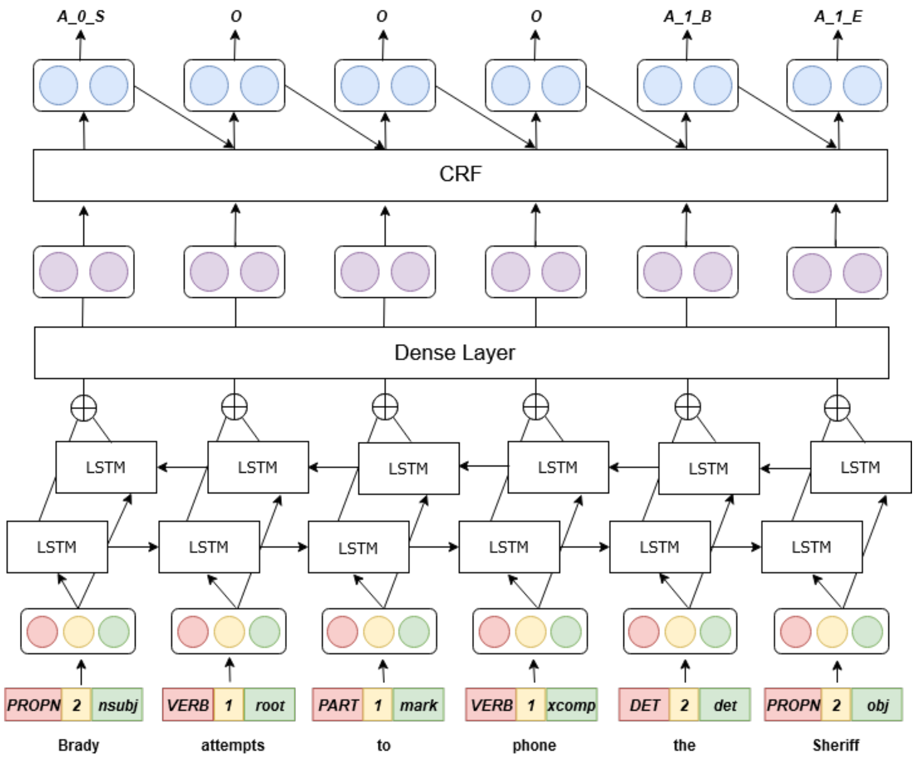

Figure 2 Architecture of the argument classifier.

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. Multiˆ2OIE: 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`.