Supervising Unsupervised Open Information Extraction Models

Arpita Roy[1], Youngja Park[2], Taesung Lee[2] and Shimei Pan[1]
[1]University of Maryland, Baltimore County
Baltimore, MD 21250, USA
{arpita2, shimei}@umbc.edu
[2] IBM T. J. Watson Research Center
Yorktown Heights, New York, USA
{young_park, taesung.lee}@ibm.com

Abstract

We propose a novel supervised open information extraction (Open IE) framework that leverages an ensemble of unsupervised Open IE systems and a small amount of labeled data to improve system performance. It uses the outputs of multiple unsupervised Open IE systems plus a diverse set of lexical and syntactic information such as word embedding, part-of-speech embedding, syntactic role embedding and dependency structure as its input features and produces a sequence of word labels indicating whether the word belongs to a relation, the arguments of the relation or irrelevant.

Comparing with existing supervised Open IE systems, our approach leverages the knowledge in existing unsupervised Open IE systems to overcome the problem of insufficient training data. By employing multiple unsupervised Open IE systems, our system learns to combine the strength and avoid the weakness in each individual Open IE system. We have conducted experiments on multiple labeled benchmark data sets. Our evaluation results have demonstrated the superiority of the proposed method over existing supervised and unsupervised models by a significant margin.

1 Introduction

Open Information Extraction (Open IE) extracts textual tuples consisting of a relation phrase and argument phrases from a sentence (Banko et al., 2007). Open IE was introduced as an alternative to traditional supervised information extraction (IE) method to address two major limitations of supervised approaches. First, supervised IE relies heavily on labeled training data. Since manual relation annotation is very expensive, this method does not scale to a large number of relations and is very difficult to adapt to new domains. Second, supervised IE systems require the target relations to be predetermined and learn to extract only the predefined relations. Therefore, they miss new and potentially meaningful domain relations that are prominent in a given dataset.

In contrast, Open IE operates in a completely domain-independent manner and is suitable when the target relations are not known in advance. Recently, Open IE has gained much attention, and various Open IE tools have been developed (Fader et al., 2011; Mausam et al., 2012; Akbik and Löser, 2012; Corro and Gemulla, 2013; Pal and Mausam, 2016; Yu et al., 2017; Kadry and Dietz, 2017; Roth et al., 2018; Stanovsky et al., 2018). Many Open IE systems demonstrated that they can scale to massive open-domain corpora such as the Web and Wikipedia (Banko et al., 2007), and the extracted tuples can be used as intermediate representation for various downstream NLP tasks such as knowledge base population (Soderland et al., 2010), question answering (Fader et al., 2014; Khot et al., 2017) and event schema induction (Mausam, 2016a).

Typically, these systems read in one sentence at a time and extract tuples with a relation phrase and one or more arguments. Most Open IE systems extract binary relations using domain-independent syntactic and lexical constraints. However, systems specialized in other syntactic constructions were also developed, such as noun-mediated relations (Pal and Mausam, 2016), n-ary relations (Akbik and Löser, 2012), nested propositions (Bhutani et al., 2016) and numerical Open IE (Saha et al., 2017a). Further, in recent years, there have been efforts to create a supervised Open IE system. (Stanovsky and Dagan, 2016) constructed an annotated corpus for Open IE, and (Stanovsky et al., 2018) and (Cui et al., 2018) used the annotated data to build a supervised Open IE system by formulating Open IE as sequence tagging and generation problems respectively.

However, while most existing Open IE systems

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 728–737, Hong Kong, China, November 3–7, 2019.

extract verbal relations, each of the systems focuses on different relational structures and extraction rules, resulting in heterogeneous results. Table 1 shows the different extraction results from the same sentence by three different Open IE systems. These variations makes it hard to compare different Open IE systems and select one for a new task, given their different strengths and weaknesses. This observation motivates us to explore an ensemble model which can learn from multiple existing Open IE systems which performs better than the underlying systems. This is especially attractive as no retraining or customization is needed to apply multiple existing Open IE systems.

In this paper, we propose a new Open IE method employing an ensemble of multiple unsupervised Open IE methods and a manually annotated data set. Similarly to (Stanovsky et al., 2018), we define Open IE as a sequence tagging problem and classify each word if it is a part of a relation, arguments or none. We first run several existing IE systems on the labeled data and use their extraction results as input features along with other rich features including word embedding, part-of-speech embedding, syntactic role embedding and syntactic dependency information. The model is then trained using the labeled data.

In this paradigm, our model can enjoy the advantages of both unsupervised Open IE approach and labeled data, since our model can learn the combined knowledge of the Open IE systems as well as optimized according to the labeled data. Evaluations with several benchmark datasets and Open IE systems show that our method outperforms the baseline systems by a large margin validating our hypothesis.

2 Supervised Ensemble of Open IE

In this work, we propose a new Open IE paradigm, a supervised ensemble of Open IE (*SenseOIE*). While there are many existing Open IE systems, each of the systems provides unique extraction rules and supports different relation constructs. This results in no single clear winner of the existing methods, when one tries to apply Open IE to a new data set. Rather, it would be more beneficial to apply multiple OpenIE systems and combine the wisdoms of all the systems. In this work we propose supervised ensemble of three different IE systems. These systems are Stanford Open IE(Angeli et al., 2015), OpenIE 5[1] and UKG (a private Open IE tool). Stanford Open IE is a dependency parser based system that uses handcrafted patterns to extract a predicate-argument triple from a sentence. On the other hand OpenIE 5 can extract verbal relation, nominal relation, relation with numeric argument and relation from consecutive sentences. UKG extracts verbal binary relations based on noun phrase detection, named entity recognition and dependency parsing. All three of these Open IE systems have different extraction rules and patterns focusing on extracting different relation tuples. Stanford Open IE, OpenIE 5 and UKG can complement each other when combined together. As a result, by ensembling these IE systems, our proposed system SenseOIE achieves better and larger coverage of all possible relation extractions. We utilize a small amount of labeled data to further optimize the model that can produce higher quality tuples.

2.1 System Overview

In this work, we consider extraction of binary relations from sentences. Let us consider an input sentence S. The goal of our system is to extract a set of relation tuples T from S, where $T = \{T_1, T_2, \ldots, T_n\}$ and the i-th tuple T_i consists of $< e_{i1}, r_i, e_{i2} >$, where r_i is the relation phrase of T_i and e_{i1} and e_{i2} are the first and second arguments of r_i. We frame this task as a sequence tagging, and the model annotates each word in the sentence to $E1$, $E2$, R or O (EOR tags). $E1$ and $E2$ denote the first and the second arguments, R is the relation, and O represents all other words. Figure 1 shows a system overview of *SenseOIE*.

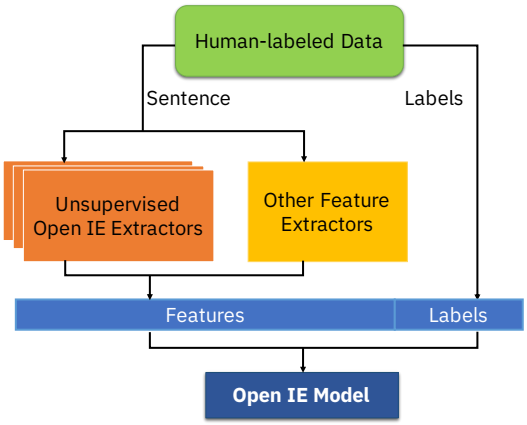

Figure 1: System Overview of *SenseOIE*

[1] http://openie.allenai.org

Input	Evil Corp has released a new improved variant of the Dridex trojan that was spread through Andromeda botnet.
OpenIE5	1. (the Dridex trojan; was spread;) 2. (Evil Corp ; has released; new variant of the Dridex trojan)
Stanford OpenIE	1. (Evil Corp; has released; variant) 2. (Evil Corp; has released; variant of Dridex trojan) 3. (Evil Corp; has released; new variant) 4. (Evil Corp; has released; new variant of Dridex trojan)
UKG	1. (Evil Corp; has released; a new improved variant of the Dridex trojan) 2. (new improved variant of the Dridex trojan; was spread through; Andromeda botnet)

Table 1: Extracted tuples by different OpenIE systems for an input sentence

2.2 Features

For an input to our system, we extract features for each word in the corpus. We first collect the Open IE system outputs by 1) running the existing Open IE systems as a black-box on the labeled corpus, 2) mapping the extracted tuples back to the original input sentence, and 3) assigning the EOR tags to each word based on the outputs of each Open IE system. This gives us k EOR tags for each word from k Open IE tools (*e.g.*, 'E, E, O' by three Open IE systems). In addition to the Open IE results, we extract part-of-speech (pos) tags, syntactic role and dependency parse tree.

In particular, we consider dependency parse tree based on the one-hop neighbors (*i.e.*, parent and children) of a word in the dependency tree. We use $\text{parent}(w_i)$, the parent of word w_i, and $\text{left-child}(w_i)$ and $\text{right-child}(w_i)$, the closest left and right children of w_i.

Formally, given an input sentence S, we extract a feature vector $\mathcal{F}(w_i)$ for each word $w_i \in S$ defined as follows:

$$\begin{aligned}\mathcal{F}(w_i) =& emb(w_i) \oplus \mathcal{F}_B(w_i) \\ &\oplus \mathcal{F}_B(\text{parent}(w_i)) \\ &\oplus \mathcal{F}_B(\text{left-child}(w_i)) \\ &\oplus \mathcal{F}_B(\text{right-child}(w_i))\end{aligned}$$

where $\oplus$ denotes concatenation, $\mathcal{F}_B(w_i) = emb(pos(w_i)) \oplus emb(role(w_i)) \oplus EOR_{1,\ldots,k}(w_i)$; $pos(w_i)$ is the part-of-speech of w_i; $role(w_i)$ is the syntactic role of w_i; $emb(\cdot)$ is the respective embedding for the categorical input that can be trained as part of the model, or pre-trained; and $EOR_{1,\ldots,k}(w_i)$ represent the k EOR tags for w_i assigned by the k Open IE tools.

2.3 Model Architecture

Our system uses bidirectional long short term memory (Bi-LSTM) (Schuster and Paliwal, 1997) to aggregate features and classify the labels of a sequence of words. The advantage of using Bi-LSTM is that we can leverage the information from neighboring words from both sides. The outputs are used in softmax for each word, producing independent probability distributions over possible EOR tags

2.4 Implementation Details

We implement *SenseOIE* using the Keras framework (Chollet et al., 2015) with TensorFlow backend [2]. We use 2 layers of stacked bidirectional LSTM, each with 100 neurons with tanh activation. We use the RMSprop optimizer which is often recommended for recurrent neural network. The model is trained using early stopping to prevent over fitting. We use the batch size of 32 samples, with 10% word-level dropout. The word embeddings are initialized using the word2vec (Mikolov et al., 2013) Google News 300-dimensions pre-trained embeddings. The part of speech and syntactic role embeddings are 25 dimensional and randomly initialized and updated during training.

3 Experiments

We validate *SenseOIE* with several benchmark data sets and compare it with the state-of-the-art Open IE systems including (1) a supervised Open IE system by (Stanovsky et al., 2018); (2) three unsupervised Open IE systems, OpenIE5 [3], Stanford OpenIE [4] (Angeli et al., 2015) and UKG which is a proprietary Open IE tool.

3.1 Baseline Systems

RnnOIE is the first supervised model built for Open IE (Stanovsky et al., 2018). The model is based on a Bi-LSTM transducer and is trained using the annotated corpus built by the same research

[2] https://www.tensorflow.org/
[3] http://openie.allenai.org
[4] https://stanfordnlp.github.io/CoreNLP/openie.html

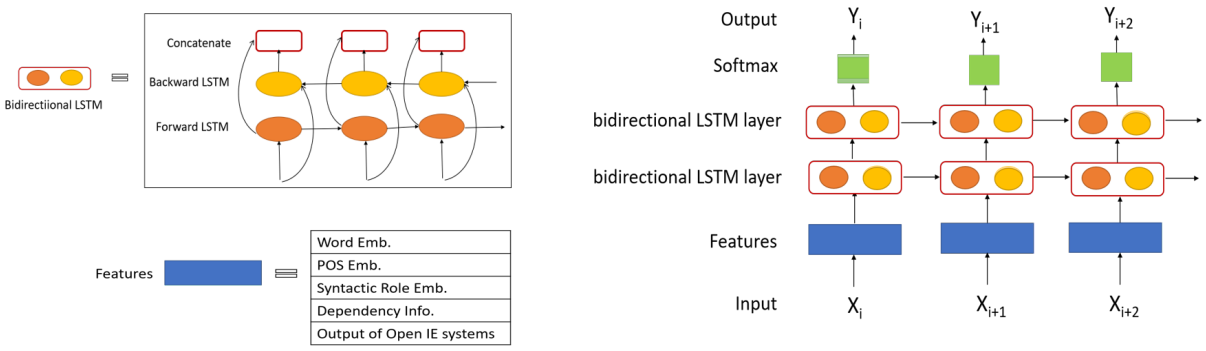

Figure 2: Model Architecture of *SenseOIE*

team (Stanovsky and Dagan, 2016). It takes a sentence and the word index of the predicate's syntactic head as input, and generates a feature vector for each word in the sentence by concatenating the word embeddings and POS tag embeddings of the word and the predicate head. Given these input features, the model learns whether the current word is part of argument of the particular predicate. At inference time, they first identify verbs and verb nominalization as candidate predicates and generate an input instance with each candidate predicate head.

Stanford Open IE is heavily based on dependency parsers. A classifier is learned for splitting a sentence into a set of logically entailed shorter clauses by recursively traversing its dependency tree and predicts whether an edge should yield an independent clause or not. In order to increase the usefulness of the extracted propositions, each self-contained clause is then maximally shortened by running natural logic inference over it. In the end, a set of 14 handcrafted patterns are used to extract a predicate-argument triple from each utterance.

OpenIE 5 is a combination of four Open IE systems CALMIE (Saha et al., 2018), BONIE (Saha et al., 2017b), RelNoun (Pal et al., 2016) and SRLIE (Christensen et al., 2011). SRLIE converts the output of a SRL system into an Open IE extraction by treating the verb as the relational phrase, and taking its role-labeled arguments as the arguments of the relation. On the other hand, RelNoun is a nominal Open IE system that extracts relations from compound noun phrases. BONIE focuses on extracting tuples where one of the arguments is a number or a quantity-unit phrase. CALMIE extracts information from conjunctive sentences by using language model based scoring and several linguistic constraints to search over hierarchical conjunct boundaries.

UKG was developed by some of this paper's authors as a tool to construct a knowledge graph for the cybersecurity domain, which contains information about cyber-incidents involving malware, campaign, and IoCs (Indicators of Compromise). It extracts verbal binary relations based on noun phrase detection, named entity recognition, dependency parsing. UKG currently extracts verbal binary relations from three dependency structures, 'NP-VP-NP', 'NP-VP-PP' and 'VP-NP-PP'. Named entity extraction is performed to detect cybersecurity-specific entities (e.g., malware names) and constrain the extractions to only cybersecurity-related relations (those with at least one argument being a cybersecurity entity). Further, UKG employs a coreference resolution, and coordination and apposition analysis to increase the recall of the extraction. To make UKG similar to other OpenIE systems for the evaluation, we did not run the cybersecurity named entity extraction but used all noun phrases as candidate arguments. Also, we did not apply the coreference resolution as other systems produce pronouns, not the referring nouns, as arguments.

Majority Votes is another baseline system that we compared with *SenseOIE*. In this system we simply take majority votes from three different IE systems that we used to generate input feature for *SenseOIE*.

3.2 Experiment Data

We use four different benchmark datasets to train and test the models. The datasets are AW-OIE (Stanovsky et al., 2018), WEB and NYT (de Sá Mesquita et al., 2013) and PENN (Xu et al., 2013). Table 2 presents more details on these datasets.

Data Set	# of Sentences	# of Tuples
AW-OIE	3,300	17,165
AW-OIE-C	3,300	13,056
WEB	500	461
NYT	222	222
PENN	100	51

Table 2: Data sets used in this work

AW-OIE corpus was created by extending the OIE2016 corpus released by (Stanovsky and Dagan, 2016). OIE2016 (Stanovsky and Dagan, 2016) was created by an automatic translation from question-answering driven semantic role labeling annotations (He et al., 2015). (Stanovsky et al., 2018) extended these techniques and apply them to the QAMR corpus (Michael et al., 2018) to create AW-OIE. This dataset is the largest dataset available for supervised open information extraction.

However, when we observe the information extracted from this dataset, we notice that the dataset is not accurate enough to be considered as a benchmark dataset. We often find missing relations and noise introduced during the automatic generation process. To solve this problem, we manually inspect the dataset and find several patterns causing this noise in the dataset. We use these patterns to filter out noisy and missing relations from the dataset and call the cleaned data set 'AW-OIE-C'.

The WEB dataset represents the challenges of dealing with web text. This contains many incomplete and grammatically unsound sentences. NYT contains formal, well written news stories from the New York Times Corpus. The PENN dataset was created from PENN Tree Bank. We use AW-OIE-C for training and testing purpose and other three datasets only for testing. We use 8,000 instances from AW-OIE-C to train *SenseOIE* and 1,456 instances to test all the models. We set aside 3,600 instances for a new experiment described in Section 3.4.

3.3 Performance Evaluation

In this section, we report the utility of our model by comparing its performance with the baseline systems on the four datasets (AW-OIE-C, WEB, NYT and PENN).

Evaluation Metric and Matching Function We compare the systems using precision, recall and F1-score. In order to compute the measures, we need to match the automated extractions by the systems and the ground truth extractions. In this work, we compute the measures based on tuple-level matching and word-level matching. Word-level matching has been used for the evaluation metric for many NER systems. For each word, we match the tag generated by the system with the word's label.

Tuple-level matching is used in other Open IE systems (Stanovsky et al., 2018; Cui et al., 2018). It is done by mapping extracted tuples with their corresponding benchmark tuples. One strategy for tuple matching would be to enforce an exact match by matching the boundaries of the extracted and benchmark tuples in text. However, as noted in earlier works (Stanovsky et al., 2018; Schneider et al., 2017), this method penalizes different but equally valid arguments, which are resulted from different annotation styles employed by different Open IE systems. Therefore, dealing with multiple OIE systems requires a less restrictive matching strategy. (Schneider et al., 2017) introduced *relaxed containment strategy*. With this strategy, extractions are counted correct as long as they contain all gold standard arguments. (Stanovsky et al., 2018) used a partial matching strategy allowing some variability (e.g., omissions of prepositions or auxiliaries) in the predicted tuples.

Following these works, we also use a partial matching strategy that allows all these kind of variabilities. We consider each argument or predicate correct, if it partially matches with the benchmark data over a certain threshold. This threshold can control the leniency or strictness of the matching function. This metric allows a more balanced and fair comparison between systems which can extract potentially correct arguments beyond benchmark extraction.

Comparison with the Baseline Systems Table 3 shows the tuple-level F1-score of *SenseOIE* and the benchmark systems. As we can see, *SenseOIE* outperforms all baseline systems with a large difference. On the AE-OIE-C dataset, *SenseOIE* achieves the highest F1-score of 0.79. In comparison with the unsupervised Open IE methods, the performance gain of *SenseOIE* ranges from 66% to 315%. *SenseOIE* outperforms OpenIE5 by 36% to 56%. When compared to UKG, *SenseOIE*'s performance gain ranges from 92% to 186%. In terms of *SenseOIE*'s performance over the different datasets, it's worth noting the differences in annotations in the different

	AW-OIE-C	Web	NYT	PENN	AW-OIE
SenseOIE	**0.79**	0.66	**0.41**	**0.52**	**0.72**
RnnOIE	-	**0.67**	0.35	0.44	0.62
OpenIE5	0.58	0.46	0.29	0.34	-
Stanford OpenIE	0.19	0.24	0.21	0.31	-
UKG	0.41	0.23	0.15	0.21	-
Majority Votes	0.40	0.42	0.24	0.27	-

Table 3: Performance (F1-score) comparison of *SenseOIE* and the baseline systems

datasets. As the test data from AW-OIE-C follows the same annotation style as the training data, the performance of *SenseOIE* is much higher on this dataset compared to other datasets.

Figure 3 shows the comparison results based on the word level F1-scores. The results also demonstrate that *SenseOIE* works better than the other systems. Especially, *SenseOIE* shows much higher accuracy in detecting words belonging to the arguments and the relation, but a slightly lower accuracy for other words.

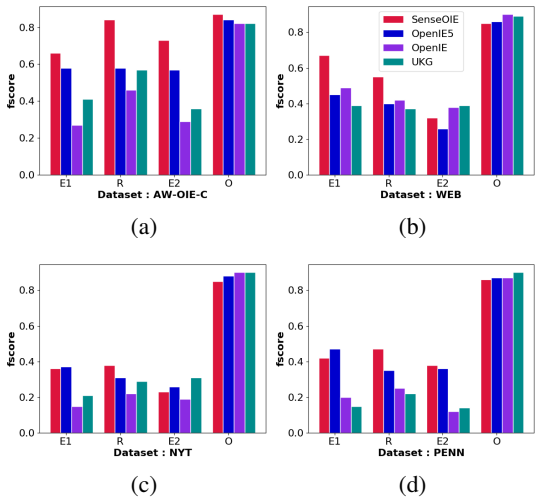

Figure 3: Word level F1-score comparison

Feature Ablation Study To investigate the contribution of each feature type on *SenseOIE*, we conducted a feature ablation study. Table 4 shows the F1-scores of several variations built with a different subset of our features. We note that the performance of *SenseOIE* is, on aevrage, 92% higher than the model using only the 'Embedding Features', which is similar to the features used in other supervised Open IE systems (Stanovsky et al., 2018; Cui et al., 2018). Overall the performance gain ranges from 13% to 156%. This performance boost proves that using an ensemble of multiple unsupervised OpenIE systems is very effective. The results also show that the ensemble of unsupervised Open IE results is more effective than the embedding features, and the combined features of the embeddings and Open IE results produce the best results. Surprisingly model without features from dependency parse tree outperforms *SenseOIE* in 2 out of 4 datasets. This might be indicator that simple concatenation is not the best way to include features from dependency tree.

3.4 *SenseOIE* as Annotator

Since *SenseOIE* outperforms the baseline systems by a large margin, we investigate if *SenseOIE* can be used to bootstrap a supervised Open IE model for new domains by automatically producing annotated data. Previously, (Cui et al., 2018) used OpenIE4 (Mausam, 2016b), an earlier version of OpenIE5, to automatically create a training dataset. The limitation of their approach is that using only one OpenIE system's extraction as ground truth will result in biased and low coverage of extracted relations. As each of the unsupervised OpenIE systems has it's own rules to extract different relations, applying only one system might miss other potential relations that can be extracted by other Open IE systems. However, since *SenseOIE* learns from multiple existing Open IE systems, it can extract many different relation types.

For this purpose, we run *SenseOIE* on the 3,600 instances from AW-OIE-C and use its extraction results as the ground truth to train a supervised model. We name this new model *SupervisedOIE* to differentiate it from *SenseOIE*. The model is quite similar to *SenseOIE* using LSTM to aggregate features and classify the labels of a sequence of words. The input features for each word are word embedding, pos embedding, syntactic role embedding, dependency tree information and label of previous word. During training label of pre-

	AW-OIE-C	Web	NYT	PENN
Embedding Features	0.42	0.58	0.16	0.27
Open IE Result Features	0.70	0.54	0.38	0.47
Embedding + Open IE Result Features	0.78	**0.69**	**0.45**	0.51
All Features	**0.79**	0.66	0.41	**0.52**

Table 4: Performance (F1-score) comparison of different feature sets. 'Embedding Features' denotes the concatenated set of word embedding, POS embedding and syntactic role embedding. 'Open IE Result Features' include only the EOR tags generated by the three unsupervised Open IE systems. 'All Features' consists of all the features as described in Section 2.2.

vious word comes from ground truth and during testing this value is predicted by the model. This feature is useful to generate multiple sequences of extractions from a single sentence. However, note that this model does not use the results of unsupervised systems as features. Figure 4 shows the architecture and features of *SupervisedOIE*.

During the inference time, to extract multiple relations from a single sentence, we use beam search to find multiple possible labels for each word. Instead of greedily choosing the most likely next step as the sequence is constructed, the beam search expands all possible next steps and keeps the k most likely results, where k is a user-specified parameter and controls the number of beams or parallel searches through the sequence of probabilities. Figure 5 shows an example of beam search predicting multiple relation extraction sequences from one sentence.

To validate the effectiveness of *SenseOIE* as an annotator model, we compare *SupervisedOIE*'s performance when trained with the human labeled data and *SenseOIE*'s extractions. As with *SenseOIE*, both models are initialized with the pre-trained word embedding and randomly initialize the part-of-speech and syntactic role embeddings. In this experiments, we set the beam size to 3 which gives an overall best performance. Table 5 shows the results from these two models. We can see that both models achieve similar F1-scores on the four test dataset. These results supports *SenseOIE*'s role as a digital annotator for unlabeled dataset.

	AW-OIE-C	Web	NYT	PENN
Human Labels	0.55	0.51	0.23	0.27
SenseOIE Labels	0.54	0.50	0.23	0.23

Table 5: Performance (F1-score) of *SupervisedOIE* trained with the human-labeled data vs. labeled data generated by *SenseOIE*

4 Related Work

Early Open IE systems apply handcrafted rules or self-supervised learning paradigm, where the extraction results which satisfy a set of syntactic constraints are considered as positive examples and the results which do not satisfy the constrains are considered as negative examples. *TextRunner* (Banko et al., 2007) is the first domain-independent Open IE system. *ReVerb* (Fader et al., 2011) extracts verbal propositions from part of speech tags using a logistic regression classifier. OLLIE (Mausam et al., 2012) is built on *ReVerb* and extracts relations from syntactic and lexical dependency patterns. *ClausIE* (Corro and Gemulla, 2013) first classifies clauses into clause types and extract tuples based on the clause type using predefined rules. Followed by these systems, Stanford OpenIE (Angeli et al., 2015) and OpenIE5 are developed by combining several different approaches as described in details in Section 3.1. Further, several systems focusing on a specialized constructs were developed, including noun-mediated relations (Pal and Mausam, 2016), n-ary relations (Akbik and Löser, 2012), nested propositions (Bhutani et al., 2016) and numerical Open IE (Saha et al., 2017a).

Recently, there have been efforts to apply deep learning methods to Open IE. *RnnOIE* (Stanovsky et al., 2018) is the first attempt to apply a supervised learning approach for Open IE using the labeled data set from (Stanovsky and Dagan, 2016) (See Section 3.1). (Cui et al., 2018) propose an encoder-decoder framework with an attention-based copying mechanism to extract binary relation tuples. They formulated Open IE as a sequence-to-sequence generation problem. Instead of relying on manually labeled data,

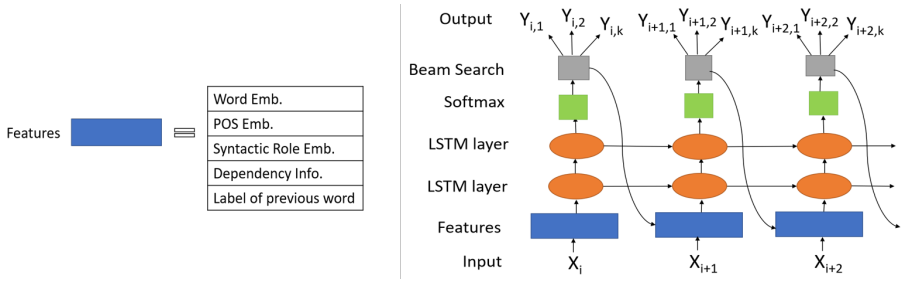

Figure 4: Model Architecture of a new *SupervisedOIE*

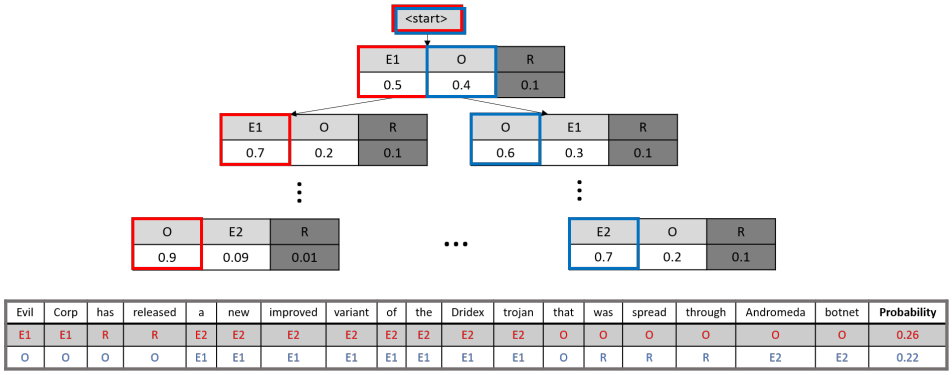

Figure 5: Example of beam search predicting multiple relation extraction sequences from one sentence

they train the model using the results of OpenIE4 (Mausam, 2016a) as labeled training data and evaluate the model using the human-labeled data from (Stanovsky and Dagan, 2016) as *RnnOIE* (Stanovsky et al., 2018). (Sun et al., 2018) present a supervised neural Open IE model for Chinese information extraction. They apply an attention-based sequence-to-sequence learning similarly to (Cui et al., 2018). However, they use the gated dependency attention mechanism based on the shortest path between a pair of words in the sentence's dependency tree. We do not compare our model with this system because it supports different target types and languages.

5 Conclusion

We propose a new Open IE paradigm which combines supervised learning and unsupervised Open IE systems. Our model uses the results of existing Open IE systems as features in addition to other linguistic features and then optimize the model using a small amount of labeled data. Validation using several benchmark data sets generated for the Open IE task shows that our method is very effective outperforming both other supervised and unsupervised Open IE systems.

Further, we investigate if our model can be applied to automatically generate annotated data to train a new supervised model for a new task. The experiment shows that a supervised model trained with the model-generated data performes similarly as the model trained with human labeled data. This result shows that our approach can overcome the cold-start problem in machine learning by leveraging existing unsupervised systems.

6 Acknowledgment

We gratefully acknowledge IBM for supporting this work.

References

Alan Akbik and Alexander Löser. 2012. Kraken: N-ary facts in open information extraction. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction, AKBC-WEKEX@NAACL-HLT 2012*, pages 52–56.

Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D. Manning. 2015. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL)*, pages 344–354.

Michele Banko, Michael J. Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2670–2676.

Nikita Bhutani, H. V. Jagadish, and Dragomir R. Radev. 2016. Nested propositions in open information extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 55–64.

François Chollet et al. 2015. Keras. https://keras.io.

Janara Christensen, Stephen Soderland, Oren Etzioni, et al. 2011. An analysis of open information extraction based on semantic role labeling. In *Proceedings of the sixth international conference on Knowledge capture*, pages 113–120. ACM.

Luciano Del Corro and Rainer Gemulla. 2013. ClausIE: clause-based open information extraction. In *22nd International World Wide Web Conference (WWW)*, pages 355–366.

Lei Cui, Furu Wei, and Ming Zhou. 2018. Neural open information extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 407–413.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1535–1545.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014. Open question answering over curated and extracted knowledge bases. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1156–1165.

Luheng He, Mike Lewis, and Luke Zettlemoyer. 2015. Question-answer driven semantic role labeling: Using natural language to annotate natural language. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 643–653.

Amina Kadry and Laura Dietz. 2017. Open relation extraction for support passage retrieval: Merit and open issues. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, pages 1149–1152.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2017. Answering complex questions using open information extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL): Short Paper*, pages 311–316.

Mausam. 2016a. Open information extraction systems and downstream applications. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4074–4077.

Mausam, Michael Schmitz, Stephen Soderland, Robert Bart, and Oren Etzioni. 2012. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 523–534.

Mausam Mausam. 2016b. Open information extraction systems and downstream applications. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 4074–4077. AAAI Press.

Julian Michael, Gabriel Stanovsky, Luheng He, Ido Dagan, and Luke Zettlemoyer. 2018. Crowdsourcing question-answer meaning representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Short Paper*, pages 560–568.

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

Harinder Pal and Mausam. 2016. Demonyms and compound relational nouns in nominal open IE. In *Proceedings of the 5th Workshop on Automated Knowledge Base Construction, AKBC@NAACL-HLT 2016*, pages 35–39.

Harinder Pal et al. 2016. Demonyms and compound relational nouns in nominal open ie. In *Proceedings of the 5th Workshop on Automated Knowledge Base Construction*, pages 35–39.

Benjamin Roth, Costanza Conforti, Nina Pörner, Sanjeev Karn, and Hinrich Schütze. 2018. Neural architectures for open-type relation argument extraction. *CoRR*, abs/1803.01707.

Filipe de Sá Mesquita, Jordan Schmidek, and Denilson Barbosa. 2013. Effectiveness and efficiency of open relation extraction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 447–457.

Swarnadeep Saha, Harinder Pal, and Mausam. 2017a. Bootstrapping for numerical open IE. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL), Short paper*, pages 317–323.

Swarnadeep Saha, Harinder Pal, et al. 2017b. Bootstrapping for numerical open ie. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 317–323.

Swarnadeep Saha et al. 2018. Open information extraction from conjunctive sentences. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2288–2299.

Rudolf Schneider, Tom Oberhauser, Tobias Klatt, Felix A. Gers, and Alexander Löser. 2017. Analysing errors of open information extraction systems. *CoRR*.

Mike Schuster and Kuldip K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Trans. Signal Processing*, 45(11):2673–2681.

Stephen Soderland, Brendan Roof, Bo Qin, Shi Xu, Mausam, and Oren Etzioni. 2010. Adapting open information extraction to domain-specific relations. *AI Magazine*, 31(3):93–102.

Gabriel Stanovsky and Ido Dagan. 2016. Creating a large benchmark for open information extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2300–2305.

Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and Ido Dagan. 2018. Supervised open information extraction. In *Proceedings of The 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT)*, pages 885–895.

Mingming Sun, Xu Li, Xin Wang, Miao Fan, Yue Feng, and Ping Li. 2018. Logician: A unified end-to-end neural approach for open-domain information extraction. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM)*, pages 556–564.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 1556–1566.

Ying Xu, Mi-Young Kim, Kevin Quinn, Randy Goebel, and Denilson Barbosa. 2013. Open information extraction with tree kernels. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 868–877.

Dian Yu, Lifu Huang, and Heng Ji. 2017. Open relation extraction and grounding. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Taipei, Taiwan, November 27 - December 1, 2017 - Volume 1: Long Papers*, pages 854–864.