Supervised Open Information Extraction

Gabriel Stanovsky*[2,3], **Julian Michael**[2], **Luke Zettlemoyer**[2], **and Ido Dagan**[1]

[1]Bar-Ilan University Computer Science Department, Ramat Gan, Israel
[2]Paul G. Allen School of Computer Science & Engineering, University of Washington, Seattle, WA
[3]Allen Institute for Artificial Intelligence, Seattle, WA
{gabis,julianjm,lsz}@cs.washington.edu
dagan@cs.biu.ac.il

Abstract

We present data and methods that enable a supervised learning approach to Open Information Extraction (Open IE). Central to the approach is a novel formulation of Open IE as a sequence tagging problem, addressing challenges such as encoding multiple extractions for a predicate. We also develop a bi-LSTM transducer, extending recent deep Semantic Role Labeling models to extract Open IE tuples and provide confidence scores for tuning their precision-recall tradeoff. Furthermore, we show that the recently released Question-Answer Meaning Representation dataset can be automatically converted into an Open IE corpus which significantly increases the amount of available training data. Our supervised model, made publicly available,[1] outperforms the state-of-the-art in Open IE on benchmark datasets.

1 Introduction

Open Information Extraction (Open IE) systems extract tuples of natural language expressions that represent the basic propositions asserted by a sentence (see Figure 1). They have been used for a wide variety of tasks, such as textual entailment (Berant et al., 2011), question answering (Fader et al., 2014), and knowledge base population (Angeli et al., 2015). However, perhaps due to limited data, existing methods use semi-supervised approaches (Banko et al., 2007; Wu and Weld, 2010), or rule-based algorithms (Fader et al., 2011; Mausam et al., 2012; Del Corro and Gemulla, 2013). In this paper, we present new data and methods for Open IE, showing that supervised learning can greatly improve performance.

Mercury filling, particularly prevalent in the USA, was banned in the EU, partly because it causes antibiotic resistance.

(mercury filling; **particularly prevalent**; in the USA)
(mercury filling; **causes**; antibiotic resistance)
(mercury filling; **was banned**; in the EU; partly because it causes antibiotic resistance)

Figure 1: Open IE extractions from an example sentence. Each proposition is composed of a tuple with a single predicate position (in bold), and an ordered list of arguments, separated by semicolons.

We build on recent work that studies other natural-language driven representations of predicate argument structure, which can be annotated by non-experts. Recently, Stanovsky and Dagan (2016) created the first labeled corpus for evaluation of Open IE by an automatic translation from question-answer driven semantic role labeling (QA-SRL) annotations (He et al., 2015). We extend these techniques and apply them to the QAMR corpus (Michael et al., 2018), an open variant of QA-SRL that covers a wider range of predicate-argument structures (Section 5). The combined dataset is the first corpus that is large and diverse enough to train an accurate extractor.

To train on this data, we formulate Open IE as a sequence labeling problem. We introduce a novel approach that can extract multiple, overlapping tuples for each sentence (Section 3), extending recent deep BIO taggers used for semantic role labeling (Zhou and Xu, 2015; He et al., 2017). We also introduce a method to calculate extraction confidence, allowing us to effectively trade off precision and recall (Section 4).

Experiments demonstrate that our approach out-

*Work performed while at Bar-Ilan University.

[1] Our code and models are made publicly available at https://github.com/gabrielStanovsky/supervised-oie

performs state-of-the-art Open IE systems on several benchmarks (Section 6), including three that were collected independently of our work (Xu et al., 2013; de Sá Mesquita et al., 2013; Schneider et al., 2017). This shows that for Open IE, careful data curation and model design can push the state of the art using supervised learning.

2 Background

In this section we survey existing Open IE systems, against which we compare our system, and available data for the task, that we will use for training and testing our model.

2.1 Different Open IE Systems and Flavors

Open IE's original goal (Banko et al., 2007) was to extend traditional (closed) information extraction, such that *a*ll of the propositions asserted by a given input sentence are extracted (see Figure 1 for examples). The broadness of this definition, along with the lack of a standard benchmark dataset for the task, prompted the development of various Open IE systems tackling different facets of the task.

While most Open IE systems aim to extract the common case of verbal binary propositions (i.e, subject-verb-object tuples), some systems specialize in other syntactic constructions, including noun-mediated relations (Yahya et al., 2014; Pal and Mausam, 2016), n-ary relations (Akbik and Löser, 2012), or nested propositions (Bhutani et al., 2016).

Many different modeling approaches have also been developed for Open IE. Some of the early systems made use of distant supervision (Banko et al., 2007; Wu and Weld, 2010), while the current best systems use rule-based techniques to extract predicate-argument structures as a post-processing step over an intermediate representation. ReVerb (Fader et al., 2011) extracts Open IE propositions from part of speech tags, OLLIE (Mausam et al., 2012), ClausIE (Del Corro and Gemulla, 2013) and PropS (Stanovsky et al., 2016) post-process dependency trees, and Open IE4[2] extracts tuples from Semantic Role Labeling (SRL) structures. These systems typically associate a confidence metric with each extraction, which allows end applications to trade off precision and recall.

2.2 Open IE Corpora

Recent work addressed the lack of labeled reference Open IE datasets for comparatively evaluating extractors. Stanovsky and Dagan (2016) created a large Open IE corpus (OIE2016) for verbal predicates by automatic conversion from QA-SRL (He et al., 2015), a variant of traditional SRL that labels arguments of verbs with simple, template-based natural language questions. Schneider et al. (2017) aggregated datasets annotated independently in previous Open IE efforts (WEB and NYT (de Sá Mesquita et al., 2013), PENN (Xu et al., 2013), and OIE2016) into a common benchmarking suite.

In addition to these, we create and make available a new Open IE training corpus, All Words Open IE (AW-OIE), derived from Question-Answer Meaning Representation (QAMR) (Michael et al., 2018), a recent extension of the QA-SRL paradigm to free-form questions over a wide range of predicate types (see Section 5). Table 1 presents more details on these datasets.

3 Task Formulation

In this work, we choose to model an Open IE proposition as a tuple consisting of a single predicate operating over a non-empty set of arguments, where the predicate and the arguments are contiguous spans from the sentence. As with traditional (binary) Open IE, every tuple should be asserted by the sentence and the order of the tuple elements should be such that it would be naturally interpretable when reading from left to right (for example, see the third tuple in Figure 1). As we show in following sections, this formulation intuitively lends itself to BIO tagging, while being expressive enough to capture a wide range of propositions.

Formally, given an input sentence S =

Dataset	Domain	#Sent.	#Tuples		
			Train	Dev	Test
AW-OIE*	Wikinews,Wiki	3300	12952	4213	-
OIE2016	News,Wiki	3200	5077	1671	1729
WEB-500	News,Web	500	-	-	461
NYT-222	News,Wiki	222	-	-	222
PENN-100	Mixed	100	-	-	51

Table 1: Datasets used in this work, following (Schneider et al., 2017). *AW-OIE (All Words Open IE) was created in the course of this work, see Section 5 for details.

[2] https://github.com/dair-iitd/OpenIE-standalone

Open IE Encoding Examples
(a) *The president claimed that he won the majority vote.*
(The president; **claimed that he won**; the majority vote)
The$_{A0-B}$ president$_{A0-I}$ claimed$_{P-B}$ that$_{P-I}$ he$_{P-I}$ won$_{P-I}$ the$_{A1-B}$ majority$_{A1-I}$ vote$_{A1-I}$
(b) *Barack Obama, a former U.S president, was born in Hawaii.*
(Barack Obama; **was born in**; Hawaii)
(a former U.S. president; **was born in**; Hawaii)
Barack$_{A0-B}$ Obama$_{A0-I}$,$_{O}$ a$_{A0-B}$ former$_{A0-I}$ U.S.$_{A0-I}$ president$_{A0-I}$,$_{O}$ was$_{P-B}$ born$_{P-I}$ in$_{P-I}$ Hawaii$_{A1-B}$
(c) *Theresa May plans for Brexit, on which the UK has voted last June.*
(the UK; **has voted on**; Brexit; last June)
Theresa$_{O}$ May$_{O}$ plans$_{O}$ for$_{O}$ Brexit$_{A1-B}$,$_{O}$ on$_{O}$ which$_{O}$ the$_{A0-B}$ UK$_{A0-I}$ has$_{P-B}$ voted$_{P-I}$ on$_{P-I}$ last$_{A2-B}$ June$_{A2-I}$

Table 2: Example sentences and respective Open IE extractions. The first line in each example presents the input pairs $(S,\ p)$, where S is the input sentence, and the predicate head, p, is denoted with an underline. Below the inputs we present the corresponding Open IE extractions. The corresponding encodings are presented below the dashed lines, where subscripts indicate the associated BIO label. Demonstrating: (a) the encoding of a multi-word predicate, (b) several arguments collapsed into the same ***A0*** argument position, (c) argument position deviating from the sentence ordering.

$(w_1, \ldots, w_n)$, a tuple consists of $(\mathbf{x_1}, \ldots, \mathbf{x_m})$, where each x_i is a contiguous subspan of S. One of the x_i is distinguished as the *predicate* (marked in bold in Figure 1), while the other spans are considered its arguments. Following this definition, we reformulate Open IE as a sequence labeling task, using a custom BIO[3] (Ramshaw and Marcus, 1995; Sang and Veenstra, 1999) scheme adapted from recent deep SRL models (He et al., 2017).

In our formulation, the set of Open IE tuples for a sentence S are grouped by predicate head-word p, as shown in Table 2. For instance, example (b) lists two tuples for the predicate head "born", which is underlined in the sentence. Grouping tuples this way allows us to run the model once for each predicate head, and accumulate the predictions across predicates to produce the final set of extractions.

Open IE tuples deviate from SRL predicate-argument structures in two major respects. First, while SRL generally deals with single-word predicates, Open IE uses multi-word predicates that often incorporate modals and embedded predicates. For example, the first tuple in the table includes the embedded predicate **claimed that he won**. Second, Open IE generates multiple extractions from a single predicate in certain syntactic constructions (e.g., apposition, co-ordination or coreference). For instance, example (b) repeats the predicate **was born in** for the two components of the apposition *Barack Obama, a former U.S. president*.

To model these unique challenges, we introduce a custom BIO tagging scheme, shown in Table 2 below the dashed lines. Predicates are encoded using the P label type, while arguments are represented using Ai labels, where i represents the argument's position within the extracted Open IE tuple. While softer than SRL's predicate-specific argument roles (e.g., ARG0), these argument positions also capture semantic information because they are arranged such that the tuple can be naturally read as a standalone statement, regardless of the complications of the source text's syntax (such as reorderings and long-distance dependencies). For instance, in the last example in Table 2, the order of the arguments in the Open IE tuple deviates from the ordering in the original sentence due to a relative clause construction (headed by the word *Brexit*).

Finally, multiple extractions per predicate are encoded by assigning the same argument index to all arguments appearing in that position across all of the predicate's extractions. For example, note that the ***A0*** argument label appears twice for the apposition in example (b). To reconstruct the extractions from the BIO labels, we produce an extraction for every possible way of choosing one argument for each index.

[3] Beginning, Inside, Outside

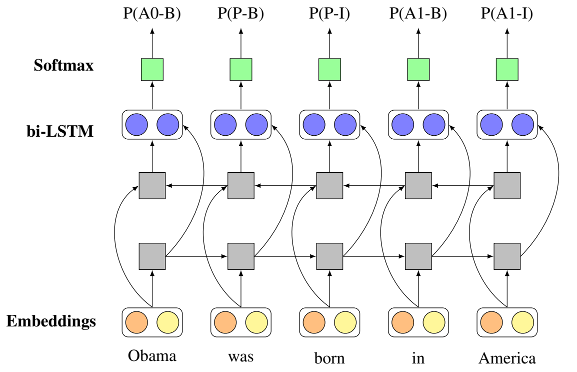


Figure 2: RNN model architecture. Orange circles represent current word features: embedding for word and part of speech. Yellow circles represent predicate features, duplicated and concatenated to all other word features.

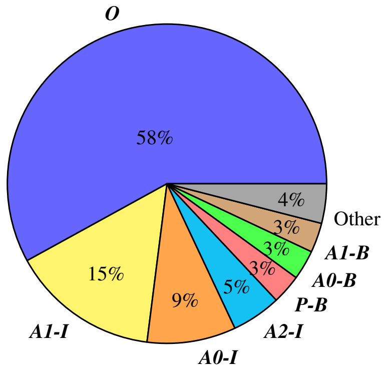


Figure 3: Word label distribution in the training set.

4 Supervised Open IE Model

Our model, named RnnOIE, is a bi-LSTM transducer, inspired by the state of the art deep learning approach for SRL suggested by Zhou and Xu (2015) and He et al. (2017). The architecture is shown in Figure 2.

Given an input instance of the form $(S,\ p)$, where S is the input sentence, and p is the word index of the predicate's syntactic head, we extract a feature vector $feat$ for every word $w_i \in S$:

$$feat(w_i,\ p) = emb(w_i) \oplus emb(pos(w_i)) \oplus \\ \oplus\, emb(w_p) \oplus emb(pos(w_p))$$

Here, $emb(w)$ is a d-dimensional word embedding, $emb(pos(w))$ is a 5-dimensional embedding of w's part of speech, and $\oplus$ denotes concatenation. We duplicate the predicate head's features on all words to allow the model to more directly access this information as it makes predicate-specific word label predictions.

The features are fed into a bi-directional deep LSTM transducer (Graves, 2012) which computes contextualized output embeddings. The outputs are used in softmaxes for each word, producing independent probability distributions over possible BIO tags.

The model is trained with gold predicate heads, using a per-word maximum likelihood objective. Figure 3 depicts the overall word label distribution within the training set. The large percentage of ***O*** labels demonstrates Open IE's tendency to shorten arguments, compared to SRL which considers full syntactic constitutes as arguments.

Inference At inference time, we first identify all verbs and nominal predicates in the sentence as candidate predicate heads. We use a Part Of Speech (POS) tagger to identify verbs, and Catvar's subcategorization frames (Habash and Dorr, 2003) for nominalizations, identifying nouns which share the same frame with a verbal equivalent (e.g., *acquisition* with *acquire*). We then generate an input instance for each candidate predicate head. For each instance, we tag each word with its most likely BIO label under the model, and reconstruct Open IE tuples from the resulting sequence according to the method described in Section 3, with the exception that we ignore malformed spans (i.e., if an ***A0-I*** label is not preceded by ***A0-I*** or ***A0-B***, we treat it as ***O***).

Assigning extraction confidence It is beneficial for an Open IE system to associate a confidence value with each predicted extraction to allow for tuning its precision-recall tradeoff. Our model does not directly produce confidence values for extractions, but it does assign probabilities to each BIO label that it predicts. We experimented with several heuristics to combine these predictions to an extraction-level confidence metric. The best performance on the development set was achieved by multiplying the probabilities of the B and I labels participating in the extraction.[4] This metric prefers shorter extractions, which correlates well with the requirements of Open IE (Bhutani et al., 2016).

Implementation details We implemented the model using the Keras framework (Chollet, 2015) with TensorFlow backend (Abadi et al., 2015). All

[4] We also tried taking the maximum or minimum observed single word-label probability.

Mercury filling, particularly prevalent in the USA, was banned in the EU, partly because it causes antibiotic resistance.			
Predicate	QA-SRL	QAMR	Open IE
made	-	What is the **filling made of**? mercury	-
prevalent	-	What was **particularly prevalent in the USA**? mercury filling	(mercury filling; **particularly prevalent**; in the USA)
banned	What was **banned**? mercury filling Where was something **banned**? the EU Why was something **banned**? partly because it causes antibiotic resistance	What was **banned in the EU partly because it causes antibiotic resistance**? mercury filling	(mercury filling; **was banned**; in the EU; partly because it causes antibiotic resistance)
causes	What **caused** something? mercury filling What did something **cause**? antibiotic resistance	What did **mercury filling cause**? antibiotic resistance	(mercury filling; **caused**; antibiotic resistance)

Table 3: Comparison of QA-SRL, QAMR, and desired Open IE annotations for an example sentence, adapted from the QAMR corpus.

hyperparameters were tuned on the `OIE2016` development set. The bi-LSTM transducer has 3 layers and each LSTM cell uses 128 hidden units and a linear rectifier (ReLU) (Nair and Hinton, 2010) activation function. The model was trained for 100 epochs in mini batches of 50 samples, with 10% word-level dropout. The word-embeddings were initialized using the GloVe 300-dimensions pre-trained embeddings (Pennington et al., 2014) and were kept fixed during training. The part of speech embeddings were randomly initialized and updated during training. Finally, we use the average perceptron part-of-speech tagger (as implemented in spaCy[5]) to predict parts of speech for input features and verb predicate identification.

5 Open IE from QAMR

This section describes our approach for automatically extracting Open IE tuples from QAMR (Michael et al., 2018), a recent extension of QA-SRL. While QA-SRL uses question templates centered on verbs, QAMR annotates free-form questions over arbitrary predicate types. The QAMR corpus consists of annotations over $5,000$ sentences. By extending the `OIE2016` training set with extractions from QAMR, we more than triple the available amount of training data.

The extraction algorithm, as well as the resulting corpus, are made publicly available at `https://github.com/gabrielStanovsky/supervised-oie`.

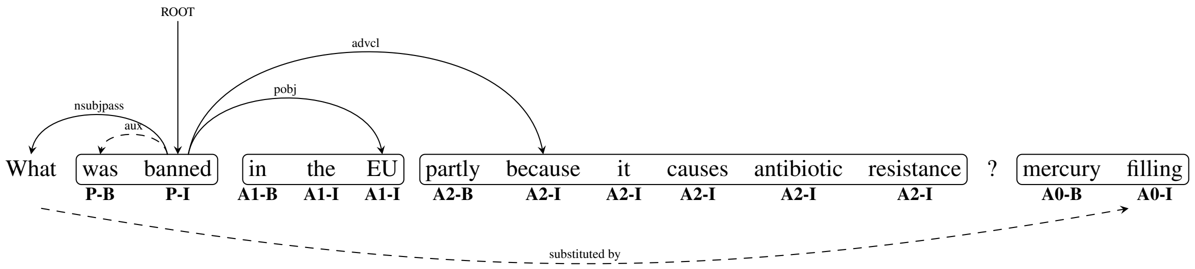
5.1 The QAMR Corpus

Question-Answer Meaning Representation, or QAMR (Michael et al., 2018), was recently proposed as an extension of QA-SRL. Like QA-SRL, QAMR represents predicate-argument structure with a set of question-answer pairs about a sentence, where each answer is a span from the sentence. However, while QA-SRL restricts questions to fit into a particular verb-centric template, QAMR is more general, allowing any natural language question that begins with a *wh*-word and contains at least one word from the sentence. This allows QAMR to express richer, more complex relations. Consider, for example, the first two entries for QAMR in Table 3. The first explicates the implicit relation **made of** from the noun compound *mercury filling*, and the second identifies the adjectival predicate **prevalent**. Neither of these can be represented in QA-SRL.

5.2 Extraction Algorithm

While QAMR's broader scope presents an opportunity to vastly increase the number and coverage of annotated Open IE tuples, it also poses additional challenges for the extraction algorithm. The free-form nature of QAMR questions means that some are over-expressive for Open IE, while in many other cases it is less obvious how to extract a predicate and a list of arguments from a question-answer pair.

[5] `https://spacy.io`



(mercury filling; **was banned**; in the EU; partly because it causes antibiotic resistance)

Figure 4: A QAMR (top) to Open IE (bottom) conversion example. The BIO labels for our encoding of the Open IE tuple appear below the text. The root of the question's dependency tree is the predicate, while its syntactic constituents are the arguments. The answer appears as the first argument of the Open IE tuple due to the passive construction.

Over-expressiveness The QAMR formalism allows many constructions that diverge from Open IE extractions, which generally are drawn verbatim from the source text. For example, the predicate **made** is introduced in the QAMR for the sentence in Table 3, despite not appearing in the sentence. To circumvent this issue, we filter out questions which: (1) introduce new content words,[6] (2) have more than one *wh*-word, (3) do not start with *who*, *what*, *when* or *where*, or (4) ask *what did X do?*, delegating the predicate to the answer.

Detecting predicates and arguments While a QA-SRL question has a designated predicate and a single argument as the answer, in QAMR, the predicate can appear anywhere in the question and its arguments are spread between the question and answer. For example, extracting an Open IE tuple for the predicate **banned** in Table 3 requires decoupling the predicate and its arguments *in the EU* and *partly because it causes antibiotic resistance*. Our solution to this problem is illustrated in Figure 4. We first run each question through a syntactic dependency parser. We then identify the predicate as the head of the question's dependency tree extended to include all dependents with an auxiliary relation (e.g., *aux*, *neg*, or *prt*). The predicted arguments are the predicate's constituent argument subtrees, while the answer to the question replaces the subtree headed by the wh-word. Finally, we employ similar heuristics to those used converting verbal QA-SRL to Open IE to find the correct argument position for the answer (Stanovsky and Dagan, 2016). For example, the passive construction in Figure 4 implies that the answer should be placed in the first argument position, while the existence of a prepositional object in, e.g., *What did he put **on the table**?* signals that the answer should be placed in the second argument position.

QAMR Open IE Tuples
(The treaty of Brussels; **was signed**; on 17 March 1948; by Belgium, the Netherlands, Luxembourg, France, and the UK)
(The treaty of Brussels; **is the precursor to**; the NATO agreement)
(The scope of publishing; **has expanded to include**; websites, blogs, and the like.)

Table 4: Tuples from the All Words Open IE Corpus, exemplifying n-ary extractions (top example), non-verbal predicates (middle), and multi-word predicates (bottom).

5.3 The All Words Open IE Corpus

As described by Michael et al. (2018), QAMR annotations were gathered via crowdsourcing in a two-stage pipeline over Wikipedia and Wikinews text. We use the training partition of the QAMR dataset, which consists of 51,063 QA pairs over 3,938 sentences. Our filtering and conversion from the QAMR corpus yields 12,952 Open IE tuples (2.5 times the size of `OIE2016`'s training corpus), composed of 7,470 (58%) verbal predicates, 4,952 (38%) nominal predicates, and 530 (4%) adjectival predicates. See Table 4 for example tuples, taken from the converted corpus.

Examining the results, we found that they are not accurate enough to constitute a gold test cor-

[6] We do not count inflected forms of verbs from the sentence, such as *caused* in the last entry of the table, as new words.

pus, partly because some relations were missed by the annotators of QAMR and partly because of noise introduced in the automatic extraction process. Instead, we use this corpus to extend the train partition of OIE2016. In the following section, we show its usefulness in significantly improving the precision and recall of our Open IE model.

6 Evaluation

We evaluate the performance of our model on the four test sets discussed in Section 2.

6.1 Experimental Setup

Metrics We evaluate each system according to three metrics. First, as is typical for Open IE, we compute a *precision-recall (PR) curve* by evaluating the systems' performance at different extraction confidence thresholds. This curve is useful for downstream applications which can set the threshold according to their specific needs (i.e., recall oriented versus precision oriented). Second, we compute the *area under the PR curve (AUC)* as a scalar measurement of the overall system performance. Finally, for each system, we report a single F1 score using a confidence threshold optimized on the development set. This can serve as a preset threshold for out-of-the-box use.

Matching function Similar to other cases in NLP, we would like to allow some variability in the predicted tuples. For example, for the sentence *The sheriff standing against the wall spoke in a very soft voice* we would want to treat both (The Sheriff; **spoke**; in a soft voice) and (The sheriff standing against the wall; **spoke**; in a very soft voice) as acceptable extractions. To that end, we follow He et al. (2015) which judge an argument as correct if and only if it includes the syntactic head of the gold argument (and similarly for predicates). For OIE2016, we use the available Penn Treebank gold syntactic trees (Marcus et al., 1993), while for the other test sets, we use predicted trees instead. While this metric may sometimes be too lenient, it does allow a more balanced and fair comparison between systems which can make different, but equally valid, span boundary decisions.

Baselines We compare our model (RnnOIE) against the top-performing systems of those evaluated most recently in Stanovsky and Dagan (2016) and in Schneider et al. (2017): Open IE4,[7] ClausIE (Del Corro and Gemulla, 2013), and PropS (Stanovsky et al., 2016).

6.2 Results

Table 5 reports the AUC and F1 scores of all of the systems on the 4 test sets. In addition, the PR curves for the two largest test sets (OIE2016 and WEB) are depicted in Figures 5a and 5b. We report results for two versions of our model: one trained on the OIE2016 training set containing only verbal predicates (*RnnOIE-verb*), and another on the extended training set that includes the automatic conversion of QAMR outlined in Section 5 (*RnnOIE-aw*).

Overall, RnnOIE-aw outperforms the other systems across the datasets. On the larger test sets (OIE2016 and WEB) it provides the best performance in terms of AUC and F1, with a superior precision-recall curve. On each of the smaller test sets, it performs best on one metric and competitively on the other.

Furthermore, on all of the test sets, extending the training set significantly improves our model's performance, showing that it benefits from the additional data and types of predicates available in the QAMR dataset. While this is most notable in the test sets which include nominalizations (WEB, NYT, and PENN), it also improves the performance on OIE2016, which is composed solely of verb predicates.

6.3 Performance Analysis

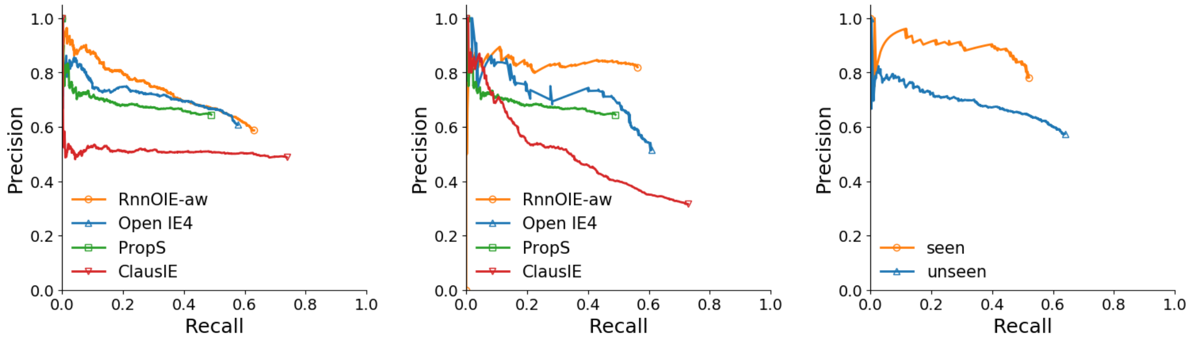
In our analysis, we find that RnnOIE generalizes to unseen predicates, produces more and shorter arguments on average than are in the gold extractions, and, like all of the systems we tested, struggles with nominal predicates.

Unseen predicates We split the propositions in the gold and predicted OIE2016 test set into two partitions, *seen* and *unseen*, based on whether the predicate head's lemma appears in the training set. The *unseen* part contains 145 unique predicate lemmas in 148 extractions, making up 24% out of the 590 unique predicate lemmas and 7% out of the 1993 total extractions in the test set. We then evaluated RnnOIE-aw on each part separately. The resulting PR curves (Figure 5c) depict overall good performance also on the unseen part, competitive with previous Open IE systems.

[7] https://github.com/dair-iitd/OpenIE-standalone

	OIE2016		WEB		NYT		PENN	
	AUC	F1 (P, R)	AUC	F1 (P, R)	AUC	F1 (P, R)	AUC	F1 (P, R)
ClausIE	.38	.59 (.49, .74)	.40	.45 (.39,.53)	.23	.30 (.24, .39)	**.28**	.34 (.24, .61)
PropS	.34	.56 (.64, .49)	.45	.59 (.44, .89)	.22	.37 (.25, .77)	**.28**	.39 (.26, .81)
Open IE4	.42	.60 (.64, .56)	.45	.56 (.63, .50)	.24	**.38 (.26, .74)**	**.28**	.43 (.37, .50)
RnnOIE-verb	.45	.59 (.57, .62)	.23	.46 (.38, .58)	.09	.25 (.20,.33)	.21	.38 (.35, .40)
RnnOIE-aw	**.48**	**.62 (.61, .64)**	**.47**	**.67 (.83, .56)**	**.25**	.35 (.24,.67)	.26	**.44 (.31,.75)**

Table 5: Performance of the OIE extractors on our test sets. Each system is tested in terms of Area Under the PR Curve (AUC), and F1 (precision and recall in parenthesis).



(a) OIE2016 (b) WEB (c) Seen vs. unseen

Figure 5: Precision-recall curves of the different OIE systems on OIE2016 (5a), WEB (5b) and seen vs. unseen predicates in RnnOIE-aw on OIE2016 (5c). See details in Section 6.

System	**# Tuples**	**Args/Prop**	**Words/Arg**
Gold	1730	2.45	5.38
ClausIE	2768	2.00	5.78
PropS	1551	2.68	5.8
Open IE4	1793	3.07	4.55
RnnOIE-aw	1993	3.19	4.68

Table 6: Output statistics of the different systems on OIE2016, versus the gold data.

This indicates that our model generalizes beyond memorization of specific predicate templates.

Argument length and number In Table 6 we compare statistics on the the outputs of the Open IE systems on OIE2016 and the gold data. The best performing systems, RnnOIE and OpenIE4, tend to produce more arguments, and each argument tends to be shorter on average, in comparison to other systems and gold.

Runtime analysis Since Open IE is intended to be usable at web scale, we timed the different Open IE systems on a batch of 3200 sentences from OIE2016, running on a Xeon 2.3GHz CPU. The results are presented in Table 8.[8] We find that our system fares well, processing only 12% fewer sentences per second than the fastest system, Open IE 4.0. Further, while these numbers are reported on CPU for the sake of fair comparison, running our neural model on a GPU (NVIDIA GeForce GTX 1080 Ti) boosts speed by a factor of more than 10 (149.25 sentences per second, on average).

Error analysis All of the systems still lack in recall across all tested corpora. We examined a random sample of 100 recall errors shared by all of the extractors across the tested datasets and found several common error types, shown in Table 7. Notably, noun and nominalized predicates still pose a challenge, appearing in 51% of the recall errors (whereas they make up 24% of all extractions). 19% of the examined errors required some

[8] PropS and ClausIE's relatively slow performance is in part due to their hard-coded use of the Stanford parser, which took on average 0.2 seconds per sentence. Using a faster parser (e.g., spaCy) may improve this performance.

Phenomenon	%	Example (sentence / gold tuple)
Noun	38	*Andre Agassi did a similar thing in his* ***hometown*** *of Las Vegas a few years ago.* (Andre Agassi; **hometown**; Las Vegas)
Sent.-level Inference	19	*John Steinbeck also* ***earned*** *alot of awards, one being the Pulitzer Prize in 1940.* (John Steinbeck; **earned**; Pulitzer Prize)
Long sentence	14	*"I don't see any radical change for our company", said David Westin, the* ***president*** *of production for Capital CitiesABC Inc. "But in the next year or so, I would expect you might see all sorts of new deals between networks and studios, joint ventures and creative financing of programs."* (David Westin; **president**; Capital Cities/ABC Inc.)
Nominalization	13	*We first heard about this when the Google-Youtube* ***acquisition*** *news broke, and wrote briefly about it here* (Google; **acquisition**; Youtube)
Noisy Informal	13	*But who knows, with Google's "****owning****" of YouTube now ..they are now in the 'media' department with that deal... so who knows if they will move on to music stuff next :P* (Google; **owning**; YouTube)
PP-attachment	10	*The novelist Franz Kafka was* ***born*** *of Jewish parentage in Prague in 1883.* (Franz Kafka; **born**; Prague)

Table 7: Analysis of frequently-occuring recall errors for all tested systems on a random sample of 100 sentences. For each phenomenon we list the percentage of sentences in which it occurs (possibly overlapping with other phenomena), and a protoypical example, taken from the WEB corpus.

	ClausIE	PropS	Open IE4	RnnOIE
CPU	4.07	4.59	**15.38**	13.51
GPU	—	—	—	**149.25**

Table 8: Runtime analysis, measured in sentences per second, of the different systems on 3200 sentences from the OIE2016 corpus on Xeon 2.3GHz CPU (top) and on an NVIDIA GeForce GTX 1080 Ti (bottom). Baselines were only run on CPU as they are currently not optimized for GPU.

form of sentence level inference, such as determining event factuality or pronoun resolution. 14% of the errors involved long sentences with over 40 words (where the average word count per sentence is 29.4).

7 Conclusions and Future Work

We present a supervised model for Open IE, formulating it as a sequence tagging problem and applying a bi-LSTM transducer to produce a state-of-the-art Open IE system. Along the way, we address several task-specific challenges, including the BIO encoding of predicates with multiple extractions and confidence estimation in our sequence tagging model. To train the system, we leverage a recently published large scale corpus for Open IE (Stanovsky and Dagan, 2016), and further extend it using a novel conversion of the QAMR corpus (Michael et al., 2018), which covers a wider range of predicates.

In addition to these contributions, this work shows that Open IE can greatly benefit from future research into the QA-SRL paradigm. For example, Open IE would directly benefit from an automatic QA-SRL extractor, while a more exhaustive or extensive annotation of QAMR would improve Open IE's performance on a wider range of predicates.

Acknowledgments

This work was supported in part by grants from the MAGNET program of the Israeli Office of the Chief Scientist (OCS); the German Research Foundation through the German-Israeli Project Cooperation (DIP, grant DA 1600/1-1); the Israel Science Foundation (grant No. 1157/16); the US NSF (IIS1252835,IIS-1562364); and an Allen Distinguished Investigator Award.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org. `http://tensorflow.org/`.

Alan Akbik and Alexander Löser. 2012. Kraken: N-ary facts in open information extraction. In *NAACL-HLT 2012: Proceedings of the The Knowledge Extraction Workshop*.

Gabor Angeli, Melvin 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 (ACL 2015)*.

Michele Banko, Michael J. Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*. pages 2670–2676.

Jonathan Berant, Ido Dagan, and Jacob Goldberger. 2011. Global learning of typed entailment rules. In *Proceedings of ACL*. Portland, OR.

Nikita Bhutani, HV Jagadish, and Dragomir Radev. 2016. Nested propositions in open information extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Austin, Texas, pages 55–64.

Franois Chollet. 2015. Keras. `https://github.com/fchollet/keras`.

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 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*. pages 447–457.

Luciano Del Corro and Rainer Gemulla. 2013. Clausie: clause-based open information extraction. In *Proceedings of the 22nd international conference on World Wide Web*. International World Wide Web Conferences Steering Committee, pages 355–366.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 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 '14, New York, NY, USA - August 24 - 27, 2014*. pages 1156–1165.

Alex Graves. 2012. Sequence transduction with recurrent neural networks. *arXiv preprint arXiv:1211.3711* .

Nizar Habash and Bonnie J. Dorr. 2003. A categorial variation database for english. In *HLT-NAACL*.

Luheng He, Kenton Lee, Mike Lewis, and Luke Zettlemoyer. 2017. Deep semantic role labeling: What works and whats next. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

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

Mitchell P Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of english: The penn treebank. *Computational linguistics* 19(2):313–330.

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*. Association for Computational Linguistics, Jeju Island, Korea, pages 523–534. `http://www.aclweb.org/anthology/D12-1048`.

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*. Association for Computational Linguistics.

Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*. pages 807–814.

Harinder Pal and Mausam. 2016. Demonyms and compound relational nouns in nominal open ie. In *AKBC@NAACL-HLT*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*. volume 14, pages 1532–1543.

Lance A Ramshaw and Mitchell P Marcus. 1995. Text chunking using transformation-based learning. *arXiv preprint cmp-lg/9505040* .

Erik F Sang and Jorn Veenstra. 1999. Representing text chunks. In *Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 173–179.

Rudolf Schneider, Tom Oberhauser, Tobias Klatt, Felix A. Gers, and Alexander Loser. 2017. Analysing errors of open information extraction systems. *CoRR* abs/1707.07499.

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)*. Association for Computational Linguistics, Austin, Texas.

Gabriel Stanovsky, Jessica Ficler, Ido Dagan, and Yoav Goldberg. 2016. Getting more out of syntax with props. *arXiv preprint* .

Fei Wu and Daniel S. Weld. 2010. Open information extraction using wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Uppsala, Sweden, pages 118–127. http://www.aclweb.org/anthology/P10-1013.

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*. Association for Computational Linguistics, Atlanta, Georgia, pages 868–877. http://www.aclweb.org/anthology/N13-1107.

Mohamed Yahya, Steven Euijong Whang, Rahul Gupta, and Alon Y. Halevy. 2014. Renoun: Fact extraction for nominal attributes. In *EMNLP*.

Jie Zhou and Wei Xu. 2015. End-to-end learning of semantic role labeling using recurrent neural networks. 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 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*. pages 1127–1137.