

OpenKI: Integrating Open Information Extraction and Knowledge Bases with Relation Inference

Dongxu Zhang^{1*}, Subhabrata Mukherjee², Colin Lockard²,

Xin Luna Dong², Andrew McCallum¹

¹University of Massachusetts Amherst

{dongxuzhang, mccallum}@cs.umass.com

²Amazon

{subhomj, clockard, lunadong}@amazon.com

Abstract

In this paper, we consider advancing web-scale knowledge extraction and alignment by integrating OpenIE extractions in the form of (subject, predicate, object) triples with Knowledge Bases (KB). Traditional techniques from universal schema and from schema mapping fall in two extremes: either they perform instance-level inference relying on embedding for (subject, object) pairs, thus cannot handle pairs absent in any existing triples; or they perform predicate-level mapping and completely ignore background evidence from individual entities, thus cannot achieve satisfying quality.

We propose *OpenKI* to handle sparsity of OpenIE extractions by performing instance-level inference: for each entity, we encode the rich information in its neighborhood in both KB and OpenIE extractions, and leverage this information in relation inference by exploring different methods of aggregation and attention. In order to handle unseen entities, our model is designed without creating entity-specific parameters. Extensive experiments show that this method not only significantly improves state-of-the-art for conventional OpenIE extractions like ReVerb, but also boosts the performance on OpenIE from semi-structured data, where new entity pairs are abundant and data are fairly sparse.

1 Introduction

Web-scale knowledge extraction and alignment has been a vision held by different communities for decades. The Natural Language Processing (NLP) community has been focusing on knowledge extraction from texts. They apply either closed information extraction according to an ontology (Mintz et al., 2009; Zhou et al., 2005), restricting to a subset of relations pre-defined in the ontology, or open information extraction (OpenIE)

to extract free-text relations (Banko et al., 2007; Fader et al., 2011), leaving the relations unaligned and thus potentially duplicated. The Database (DB) community has been focusing on aligning relational data or WebTables (Cafarella et al., 2008) by schema mapping (Rahm and Bernstein, 2001), but the quality is far below adequate for assuring correct data integration.

We propose advancing progress in this direction by applying *knowledge integration* from OpenIE extractions. OpenIE extracts SPO (subject, predicate, object) triples, where each element is a text phrase, such as **E1**: (“Robin Hood”, “Full Cast and Crew”, “Leonardo Decaprio”) and **E2**: (“Ang Lee”, “was named best director for”, “Brokeback”). OpenIE has been studied for text extraction extensively (Yates et al., 2007; Fader et al., 2011; Mausam et al., 2012), and also for semi-structured sources (Bronzi et al., 2013), thus serves an effective tool for web-scale knowledge extraction. The remaining problem is to align text-phrase predicates¹ from OpenIE to knowledge bases (KB). Knowledge integration answers the following question: given an OpenIE extraction (s, p, o), how can one populate an existing KB using relations in the pre-defined ontology?

The problem of knowledge integration is not completely new. The DB community has been solving the problem using schema mapping techniques, identifying mappings from a source schema (OpenIE extractions in our context) to a target schema (KB ontology in our context) (Rahm and Bernstein, 2001). Existing solutions consider *predicate-level* (i.e., attribute) similarity on names, types, descriptions, instances, and so on, and generate mappings like “email” mapped to “email-

¹We also need to align text-phrase entities, which falls in the area of entity linking (Dredze et al., 2010; Ji et al., 2014); it is out of scope of this paper and we refer readers to relevant references.

* This work was performed while at Amazon.

address”; “first name” and “last name” together mapped to “full name”. However, for our example “Full Cast and Crew”, which is a union of multiple KB relations such as “directed.by”, “written.by”, and “actor”, it is very hard to determine a mapping at the predicate level.

On the other hand, the NLP community has proposed *Universal Schema* (Riedel et al., 2013) to apply *instance-level* inference from both OpenIE extractions and knowledge in existing knowledge bases: given a set of extractions regarding an entity pair (s, o) and also information of each entity, infer new relations for this pair. One drawback of this method is that it cannot handle unseen entities and entity pairs. Also, the technique tends to overfit when the data is sparse due to large number of parameters for entities and entity pairs. Unfortunately, in the majority of the real extractions we examined in our experiments, we can find only 1.4 textual triples on average between the subject and object. The latest proposal *Rowless Universal Schema* (Verga et al., 2017) removes the entity-specific parameters and makes the inference directly between predicates and relations, thereby allowing us to reason about unseen entity pairs. However, it completely ignores the entities themselves, so in a sense falls back to *predicate-level* decisions, especially when only one text predicate is observed.

In this paper we propose a solution that leverages information about the individual entities whenever possible, and falls back to predicate-level decisions only when both involved entities are new. Continuing with our example **E1** – if we know from existing knowledge that “Leonardo” is a famous actor and has rarely directed or written a movie, we can decide with a high confidence that this predicate maps to @film.actor in this triple, even if our knowledge graph knows nothing about the new movie “Robin Hood”. In particular, we make three contributions in this paper.

1. We design an embedding for each entity by exploring rich signals from its neighboring relations and predicates in KB and OpenIE. This embedding provides a soft constraint on which relations the entities are likely to be involved in, while keeping our model free from creating new entity-specific parameters so allowing us to handle unseen entities during inference.
2. Inspired by predicate-level mapping from schema mapping and instance-level inference

from universal schema, we design a joint model that leverages the neighborhood embedding of entities and relations with different methods of aggregation and attention.

3. Through extensive experiments on various OpenIE extractions and KB, we show that our method improves over state-of-the-arts by 33.5% on average across different datasets.

In the rest of the paper, we define the problem formally in Section 2, present our method in Section 3, describe experimental results in Section 4, and discuss related work in Section 5.

2 Problem Overview

Problem Statement. Given (i) an existing knowledge base KB of triples (s, p, o) – where $s, o \in E^{KB}$ (set of KB entities) and $p \in R^{KB}$ (set of KB relations), and (ii) a set of instances (s', p', o') from OpenIE extraction (s' and o' may not belong to E^{KB} , and p' are text predicates)²: predict $score(s', p, o')$ – where $p \in R^{KB}$.

For example, given **E1** and **E2** as OpenIE extractions and background knowledge bases (KB) like IMDB, we want to predict “@film.actor” relation given **E1** and “@film.directed.by” relation given **E2** as the target KB relations between the participating entities. Particularly, we want to perform this relation inference at *instance-level*, which can be different for different entities sharing the same predicate. Table 1 introduces important notations used in this paper.

s	Subject
o	Object
p	KB relation or text predicate
$v_s, v_p, v_o, v_{s,o}$	Embedding vectors of s, p, o and (s, o)
$S(s, p, o)$	Scoring function for (s, p, o) to be true
$Agg_{p \in R(s,o)}(v_p)$	Aggregation function over embeddings (v_p) of p shared by s and o .

Table 1: Notation table.

2.1 Existing Solution and Background

Universal Schema (F-Model) (Riedel et al., 2013) is modeled as a matrix factorization task where entity pairs, e.g., (*RobinHood*, *Leonardo Decaprio*) form the rows, and relations from OpenIE and KB form the columns (e.g., @film.actor, “Full Cast and Crew”). During training, we observe some positive entries in the matrix and the objective is to predict the missing cells at test time.

² In this paper, a ‘relation’ always refers to a KB relation, whereas a ‘predicate’ refers to an OpenIE textual relation.

Each (subject, predicate, object) triple is scored as:

$$S_F(s, p, o) = v_{s,o} \cdot v_p^T$$

where, $v_{s,o} \in \mathcal{R}^d$ is the embedding vector of the entity pair (subject, object), v_p is the embedding vector of a KB relation or OpenIE predicate, and the triple score is obtained by their dot product. The parameters v_p and $v_{s,o}$ are randomly initialized and learned via gradient descent.

One of the drawbacks of universal schema is the explicit modeling of entity pairs using free parameters $v_{s,o}$. Therefore, it cannot model unseen entities. This also makes the model overfit on our data as the number of OpenIE text predicates observed with each entity pair is rather small (1.4 on average in our datasets).

Universal Schema (E-Model) (Riedel et al., 2013) considers entity-level information, thus decomposing the scoring function from the F-model as follows:

$$\begin{aligned} S_E(s, p, o) &= S_{subj}(s, p) + S_{obj}(p, o) \\ &= v_s \cdot v_p^{subjT} + v_o \cdot v_p^{objT} \end{aligned} \quad (1)$$

where each relation is represented by two vectors corresponding to its argument type for a subject or an object. The final score is an additive summation over the subject and object scores S_{subj} and S_{obj} that implicitly contain the argument type information of the predicate p . Thus, a joint F- and E-model of $S_{F+E} = S_F + S_E$ can perform relation inference at *instance-level* considering the entity information. Although the E-model captures rich information about entities, it still cannot deal with unseen entities due to the entity-specific free parameters v_s and v_o .

Rowless Universal Schema (Rowless) (Verga et al., 2017) handles new entities as follows. It considers all relations in KB and OpenIE that the subject s and object o co-participates in (denoted by $R(s, o)$), and represents the entity pair with an aggregation over embeddings of these relations.

$$\begin{aligned} v_{s,o}^{Rowless} &= Agg_{p' \in R(s,o)}(v_{p'}) \\ S_{Rowless}(s, p, o) &= v_{s,o}^{Rowless} \cdot v_p^T \end{aligned} \quad (2)$$

$Agg(\cdot)$ is an aggregation function like average pooling, max pooling, hard attention (Rowless MaxR) or soft attention given query relations (Rowless Attention) (Verga et al., 2017). The

Rowless model ignores the individual information of entities, and therefore falls back to making *predicate-level* decisions in a sense, especially when there are only a few OpenIE predicates for an entity pair.

3 Our Approach

We propose *OpenKI* for *instance-level* relation inference such that it (i) captures rich information about each entity from its neighborhood KB relations and text predicates to serve as background knowledge and generalizes to unseen entities by *not* learning any entity-specific parameters (only KB relations and OpenIE predicates are parameterized) (ii) considers both shared predicates and entity neighborhood information to encode entity pair information. Figure 1 shows the architecture of our model.

3.1 Entity Neighborhood Encoder (ENE)

The core of our model is the Entity Neighborhood Encoder. Recall that Rowless Universal Schema represents each entity pair with common relations shared by this pair. However, it misses critical information when entities do not only occur in the current entity pair, but also interact with other entities. This *entity neighborhood* can be regarded as a soft and fine-grained entity type information that could help infer relations when observed text predicates are ambiguous (polysemous), noisy (low quality of data source) or low-frequency (sparsity of language representation).³

Our aim is to incorporate this entity neighborhood information into our model for *instance-level* relation inference while keeping it free of entity-specific parameters. To do this, for each entity, we leverage all its *neighboring KB relations and OpenIE predicates* for relation inference. We aggregate their embeddings to obtain two scores for the subject and object separately in our ENE model. The subject score S_{subj}^{ENE} for an entity considers the aggregated embedding of its participating KB relations and OpenIE predicates where it serves as a subject (similar for the object score S_{obj}^{ENE}):

³Note that, the notion of *entity neighborhood* is different from the *Neighborhood* model in the Universal Schema work (Riedel et al., 2013). Our *entity neighborhood* captures information of each entity, whereas their Neighborhood model leverages prediction from similar predicates.

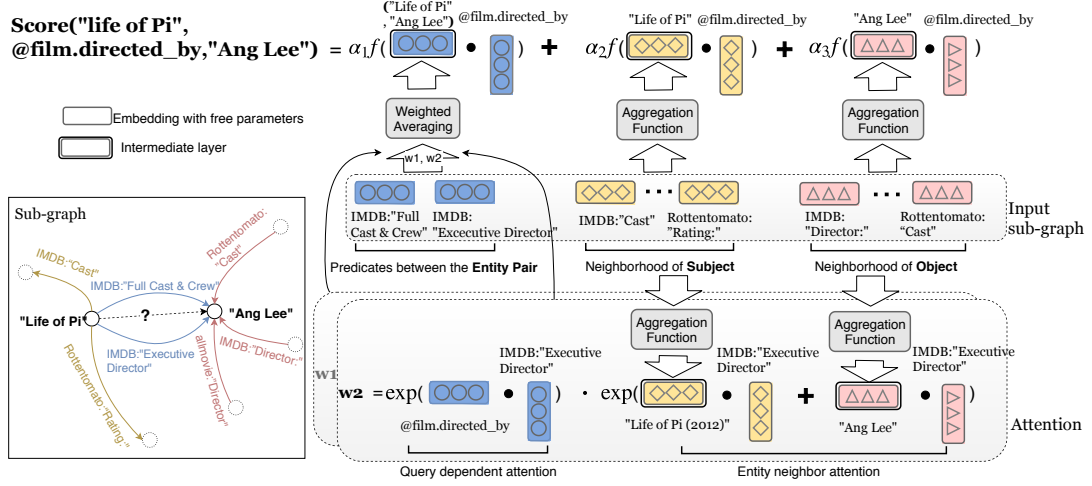


Figure 1: Architecture of the proposed method. In this example, the ENE model uses “Ang Lee’s” neighboring predicates “IMDB:Director” and “allmovie:Director” for predicting the target KB relation “@film.directed.by”. The attention mechanism assigns a larger weight over “IMDB:Executive Director” for generating entity pair embedding. Different colors of vectors represent different sets of parameters. The Entity Neighborhood Encoder (ENE) (square & triangle fillers) model contributes the following components to the final scoring function: (1) entity neighborhood scores S_{subj}^{ENE} and S_{obj}^{ENE} of the subject and object respectively; (2) neighborhood signal for the attention module to calculate the weight of each text predicate and the attention score $S^{Att}(s, p, o)$ (circle filler).

$$\begin{aligned}
 v_{subj}^{agg} &= \text{Agg}_{p' \in R(s, \cdot)}(v_{p'}^{subj}) \\
 S_{subj}^{ENE}(s, p) &= v_{subj}^{agg} \cdot v_p^{subj T} \\
 v_{obj}^{agg} &= \text{Agg}_{p' \in R(\cdot, o)}(v_{p'}^{obj}) \\
 S_{obj}^{ENE}(p, o) &= v_{obj}^{agg} \cdot v_p^{obj T}
 \end{aligned} \quad (3)$$

$R(s, \cdot)$ denotes all neighboring relations and predicates of the subject s (similar for the object). v_p^{subj} and v_p^{obj} are the only free parameters in ENE. These are randomly initialized and then learned via gradient descent. We choose average pooling as our aggregation function to capture the proportion of different relation and predicate types within the target entity’s neighborhood.

3.2 Attention Mechanism

Given multiple predicates between a subject and an object, only some of them are important for predicting the target KB relation between them. For example, in Figure 1, the predicate “Executive Director” is more important than “Full Cast & Crew” to predict the KB relation “@film.directed.by” between “Life of Pi” and “Ang Lee”.

We first present a query-based attention mechanism from earlier work, and then present our own solution with a neighborhood attention and combining both in a dual attention mechanism.

3.2.1 Query Attention

The first attention mechanism uses a query relation q (i.e., the target relation we may want to predict) to find out the importance (weight) $w_{p|q}$ of different predicates p with respect to q with v_p and v_q as the corresponding relation embeddings.

$$w_{p|q} = \frac{\exp(v_q \cdot v_p^T)}{\sum_{p'} \exp(v_q \cdot v_{p'}^T)}$$

Thus, given each query relation q , the model tries to find evidence from predicates that are most relevant to the query. Similar techniques have been used in (Verga et al., 2017). We can also use hard attention (referred as MaxR) instead of soft attention where the maximum weight is replaced with one and others with zero. One potential shortcoming of this attention mechanism is its sensitivity to noise, whereby it may magnify sparsely observed predicates between entities.

3.2.2 Neighborhood Attention

In this attention mechanism, we use the subject and object’s neighborhood information as a filter to remove unrelated predicates. Intuitively, the entity representation generated by the ENE from its neighboring relations can be regarded as a soft and fine-grained entity type information.

Consider the embedding vectors v_{subj}^{agg} and v_{obj}^{agg} in Equation 3 that are aggregated from the entity’s neighboring predicates and relations using

an aggregation function. We compute the similarity $w_{p|Nb}$ between an entity’s neighborhood information given by the above embeddings and a text predicate p to enforce a soft and fine-grained argument type constraint over the text predicate:

$$w_{p|Nb} = \frac{\exp(v_{subj}^{agg} \cdot v_p^{subjT} + v_{obj}^{agg} \cdot v_p^{objT})}{\sum_{p'} \exp(v_{subj}^{agg} \cdot v_{p'}^{subjT} + v_{obj}^{agg} \cdot v_{p'}^{objT})}$$

Finally, we combine both the query-dependent and neighborhood-based attention into a *Dual Attention* mechanism:

$$w_{p|q+Nb} = w_{p|q} \cdot w_{p|Nb}$$

$$w_p = \frac{w_{p|q+Nb}}{\sum_{p'} w_{p'|q+Nb}}$$

And the score function is given by:

$$S^{Att}(s, q, o) = Agg_{p \in R(s,o)}(v_p) \cdot v_q^T$$

$$= \left(\sum_p w_p v_p \right) \cdot v_q^T \quad (4)$$

3.3 Joint Model: OpenKI

All of the above models capture different types of features. Given a target triple (s, p, o) , we combine scores from Eq. 3 and Eq. 4 in our final OpenKI model. It aggregates the neighborhood information of s and o and also uses an attention mechanism to focus on the important predicates between s and o . Refer to Figure 1 for an illustration. The final score of (s, p, o) is given by:

$$score(s, p, o) = f_1(S^{Att}(s, p, o)) * ReLU(\alpha_1)$$

$$+ f_2(S_{subj}^{ENE}(s, p)) * ReLU(\alpha_2)$$

$$+ f_3(S_{obj}^{ENE}(p, o)) * ReLU(\alpha_3)$$

where $f_i(X) = \sigma(a_i X + b_i)$ normalizes different scores to a comparable distribution. $ReLU(\alpha_i)$ enforces non-negative weights that allow scores to only contribute to the final model without canceling each other. a_i, b_i, α_i are free parameters that are learned during the back propagation gradient descent process.

3.4 Training Process

Our task is posed as a ranking problem. Given an entity pair, we want the observed KB relations between them to have higher scores than the unobserved ones. Thus, a pair-wise ranking based loss function is used to train our model:

$$L(s, p_{pos}, p_{neg}, o) = \max(0, \gamma - score(s, p_{pos}, o) + score(s, p_{neg}, o))$$

where p_{pos} refers to a positive relation, p_{neg} refers to a uniformly sampled negative relation, and γ is the margin hyper-parameter. We optimize the loss function using Adam (Kingma and Ba, 2014). The training process uses early stop according to the validation set.

3.5 Explicit Argument Type Constraint

Subject and object argument types of relations help in filtering out a large number of candidate relations that do not meet the argument type, and therefore serve as useful constraints for relation inference. Similar to (Yu et al., 2017), we identify the subject and object argument type of each relation by calculating its probability of co-occurrence with subject / object entity types. During inference, we select candidate relations by performing a post-processing filtering step using the subject and object’s type information when available.

4 Experiments

4.1 Data

We experiment with the following OpenIE datasets and Knowledge Bases.

(i) Ceres (Lockard et al., 2019) works on semi-structured web pages (e.g., IMDB) and exploits the DOM Tree and XPath (Olteanu et al., 2002) structure in the page to extract triples like (*Incredibles 2*, *Cast and Crew*, *Brad Bird*) and (*Incredibles 2*, *Writers*, *Brad Bird*). We apply Ceres on the SWDE (Hao et al., 2011) movie corpus to generate triples. We align these triples to two different knowledge bases: (i) IMDB and (ii) subset of Freebase with relations under */film* domain. The average length of text predicates is 1.8 tokens for Ceres extractions.

(ii) ReVerb (Fader et al., 2011) works at sentence level and employs various syntactic constraints like part-of-speech-based regular expressions and lexical constraints to prune incoherent and uninformative extractions. We use 3 million ReVerb extractions from ClueWeb where the subject is already linked to Freebase (Lin et al., 2012)⁴. We align these extractions to (i) entire Freebase and (ii) subset of Freebase with relations under */film* domain. The average length of text predicates is 3.4 tokens for ReVerb extractions.

⁴Extractions are downloadable at http://knowitall.cs.washington.edu/linked_extractions/

	ReVerb + Freebase	ReVerb + Freebase(/film)	Ceres + Freebase(/film)	Ceres + IMDB
Training set				
# entity pairs for model training	40,878	1,102	23,389	64,539
# KB relation types	250	64	54	66
# OpenIE predicate types	124,836	35,366	124	178
Test set				
# test triples	4938	402	986	998
Avg./Med. # text edges per entity pair	1.74 / 1	1.49 / 1	1.35 / 1	1.23 / 1
Avg./Med. # edges for each subj	95.71 / 9	100.27 / 23	48.80 / 44	121.06 / 110
Avg./Med. # kb edges for each subj	61.41 / 4	8.30 / 6	7.00 / 7	30.77 / 29
Avg./Med. # edges for each obj	699.89 / 62	24.81 / 8	558.33 / 9	775.74 / 12
Avg./Med. # kb edges for each obj	325.70 / 23	10.11 / 6	340.96 / 3	606.31 / 6

Table 2: Data Statistics (Avg: Average, Med: Median).

Models	ReVerb + Freebase	ReVerb + Freebase(/film)	Ceres + Freebase (/film)	Ceres + IMDB
$P(p \bar{p}')$ (similar to PMI (Angeli et al., 2015))	0.412	0.301	0.507	0.663
$P(p s, \bar{p}', o)$	0.474	0.317	0.627	0.770
E-model (Riedel et al., 2013)	0.215	0.156	0.431	0.506
ENE	0.479	0.359	0.646	0.808
Rowless with MaxR (Verga et al., 2017)	0.318	0.285	0.481	0.659
Rowless with Query Attn. (Verga et al., 2017)	0.326	0.278	0.512	0.695
OpenKI with MaxR	0.500	0.378	0.649	0.802
OpenKI with Query Att.	0.497	0.372	0.663	0.800
OpenKI with Neighbor Att.	0.495	0.372	0.650	0.813
OpenKI with Dual Att.	0.505	0.365	0.658	0.814

Table 3: Mean average precision (MAP) of different models over four data settings.

In order to show the generalizability of our approach to traditional (non OpenIE) corpora, we also perform experiments in the New York Times (NYT) and Freebase dataset (Riedel et al., 2010), which is a well known benchmark for distant supervision relation extraction. We consider the sentences (average length of 18.8 tokens) there to be a proxy for text predicates. These results are presented in Section 4.5.

Data preparation: We collect all entity mentions M from OpenIE text extractions, and all candidate entities E_{KB} from KB whose name exists in M . We retain the sub-graph G_{KB} of KB triples where the subject and object belongs to E_{KB} . Similar to (Riedel et al., 2013), we use string match to collect candidate entities for each entity mention. For each pair of entity mentions, we link them if two candidate entities in E_{KB} share a relation in KB. Otherwise, we link each mention to the most common candidate. For entity mentions that cannot be linked to KB, we consider them as new entities and link together mentions that share same text .

For validation and test, we randomly hold-out a part of the entity pairs from G_{KB} where text pred-

icates are observed. Our training data consists of the rest of G_{KB} and all the OpenIE text extractions. In addition, we exclude direct KB triples from training where corresponding entity pairs appear in the test data (following the data setting of (Toutanova et al., 2015)). Table 2 shows the data statistics ⁵.

We adopt a similar training strategy as Universal Schema for the Ceres dataset – that not only learns direct mapping from text predicates to KB relations, but also clusters OpenIE predicates and KB relations by their co-occurrence. However, for the ReVerb data containing a large number of text predicates compared to Ceres, we only learn the direct mapping from text predicates to KB relations that empirically works well for this dataset.

4.2 Verifying Usefulness of Neighborhood Information: Bayesian Methods

To verify the usefulness of the entity’s neighborhood information, we devise simple Bayesian methods as baselines. The simplest method counts

⁵Our datasets with train, test, validation split are downloadable at <https://github.com/zhangdongxu/relation-inference-naacl19> for benchmarking.

the co-occurrence of text predicates and KB relations (by applying Bayes rule) to find the conditional probability $P(p|\bar{p}')$ of a target KB relation p given a set of observed text predicates \bar{p}' . This performs relation inference at *predicate-level*.

Then, we can include the entity’s relational neighbors in the Bayesian network by adding the neighboring predicates and relations of the subject (given by p_s^N) and object (given by p_o^N) to find $P(p|s, \bar{p}', o)$, which performs relation inference at the *instance-level*. The graph structures of these three Bayesian methods are shown in Figure 2. For detailed formula derivation, please refer to Appendix A.1.

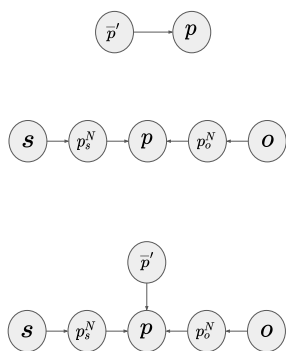


Figure 2: Structures of $P(p|\bar{p}')$, $P(p|s, o)$, $P(p|s, \bar{p}', o)$ are listed from top to bottom.

4.3 Baselines and Experimental Setup

Angeli et al. (2015) employ point-wise mutual information (PMI) between target relations and observed predicates to map OpenIE predicates to KB relations. This is similar to our Bayes conditional probability $P(p|\bar{p}')$. This baseline operates at *predicate-level*. To indicate the usefulness of entity neighborhood information, we also compare with $P(p|s, \bar{p}', o)$ as mentioned in Section 4.2. For the advanced embedding-based baselines, we compare with the E-model and the Rowless model (with MaxR and query attention) introduced in Section 2.1.

Hyper-parameters: In our experiments, we use 25 dimensional embedding vectors for the Rowless model, and 12 dimensional embedding vectors for the E- and ENE models. We use a batch-size of 128, and 16 negative samples for each positive sample in a batch. Due to memory constraints, we sample at most 8 predicates between entities and 16 neighbors for each entity during training. We use $\gamma = 1.0$ and set the learning rate to $5e-3$ for ReVerb and $1e-3$ for Ceres datasets.

Evaluation measures: Our task is a multi-label

task, where each entity pair can share multiple KB relations. Therefore, we consider each KB relation as a query and compute the Mean Average Precision (MAP) – where entity pairs sharing the query relation should be ranked higher than those without the relation. In Section 4.4 we report MAP statistics for the 50 most common KB relations for ReVerb and Freebase dataset, and for the 10 most common relations in other domain specific datasets. The left out relations involve few triples to report any significant statistics. We also report the area under the precision recall curve (AUC-PR) for evaluation in Section 4.5.

4.4 Results

Table 3 shows that the overall results. OpenKI achieves significant performance improvement over all the baselines. Overall, we observe 33.5% MAP improvement on average across different datasets.

From the first two rows of Table 3, we observe the performance to improve as we incorporate neighborhood information into the Bayesian method. This depicts the strong influence of the entity’s neighboring relations and predicates for relation inference.

The results show that our Entity Neighbor Encoder (ENE) outperforms the E-Model significantly. This is because the majority of the entity pairs in our test data have *at least one unseen entity* (refer to Table 4), which is very common in the OpenIE setting. The E-model cannot handle unseen entities because of its modeling of entity-specific parameters. This demonstrates the benefit of encoding entities with their neighborhood information (KB relations and text predicates) rather than learning entity-specific parameters. Besides, ENE outperforms the Rowless Universal Schema model, which does not consider any information surrounding the entities. This becomes a disadvantage in sparse data setting where only a few predicates are observed between an entity pair.

Finally, the results also show consistent improvement of OpenKI model over only-Rowless and only-ENE models. This indicates that the models are complementary to each other. We further observe significant improvements by applying different attention mechanisms over the OpenKI MaxR model – thus establishing the effectiveness of our attention mechanism.

Unseen entity: Table 4 shows the data statistics

of unseen entity pairs in our test data. The most common scenario is that only one of the entity in a pair is observed during training, where our model benefits from the extra neighborhood information of the observed entity in contrast to the Rowless model.

Dataset	Both seen	One unseen	Both unseen
ReVerb + Freebase	864	3232	842
ReVerb + Freebase(/film)	27	147	228
Ceres + Freebase(/film)	383	561	42
Ceres + IMDB	462	533	3

Table 4: Statistics for unseen entities in test data. “Both seen” indicates both entities exist in training data; “One unseen” indicates only one of the entities in the pair exist in training data; “Both unseen” indicates both entities were unobserved during training.

Models	All data	At least one seen
Rowless Model	0.278	0.282
OpenKI with Dual Att.	0.365	0.419

Table 5: Mean average precision (MAP) of Rowless and OpenKI on ReVerb + Freebase (/film) dataset.

Table 5 shows the performance comparison on test data where at least one of the entity is known at test time. We choose ReVerb+Freebase(/film) for analysis because it contains the largest proportion of test triples where both entities are unknown during training. From the results, we observe that OpenKI outperforms the Rowless model by 48.6% when at least one of the entity in the triple is observed during training. Overall, we obtain 31.3% MAP improvement considering all of the test data. This validates the efficacy of encoding entity neighborhood information where at least one of the entities is known at test time. In the scenario where both entities are unknown at test time, the model falls back to the Rowless setting.

Models	MAP
Rowless Model	0.695
+Type Constraint	0.769 \uparrow 10.6%
ENE Model	0.808
+Type Constraint	0.818 \uparrow 1.2%
OpenKI with Dual Att.	0.814
+Type Constraint	0.828 \uparrow 1.7%

Table 6: MAP improvement with argument type constraints on Ceres + IMDB dataset.

Explicit Argument Type Constraint: As discussed in Section 3.5, incorporating explicit type constraints can improve the model performance.

However, entity type information and argument type constraints are not always available especially for new entities. Table 6 shows the performance improvement of different models with entity type constraints. We observe the performance improvement of the ENE model to be much less than that of the Rowless model with explicit type constraint. This shows that the ENE model already captures soft entity type information while modeling the neighborhood information of an entity in contrast to the other methods that require explicit type constraint.

4.5 Results on NYT + Freebase Dataset

Prior works (Surdeanu et al., 2012; Zeng et al., 2015; Lin et al., 2016; Qin et al., 2018) on distantly supervised relation extraction performed evaluations on the New York Times (NYT) + Freebase benchmark data developed by Riedel et al. (2010)⁶. The dataset contains sentences whose entity mentions are annotated with Freebase entities as well as relations. The training data consists of sentences from articles in 2005-2006 whereas the test data consists of sentences from articles in 2007. There are 1950 relational facts in our test data⁷. In contrast to our prior experiments in the semi-structured setting with text predicates, in this experiment we consider the sentences to be a proxy for the text predicates.

Models	AUC-PR
PCNN + MaxR (Zeng et al., 2015)	0.325
PCNN + Att. (Lin et al., 2016)	0.341
ENE	0.421
OpenKI with Dual Att.	0.461

Table 7: Performances on NYT + Freebase data.

Table 7 compares the performance of our model with two state-of-the-art works (Zeng et al., 2015; Lin et al., 2016) on this dataset using AUC-PR as the evaluation metric.

Overall, OpenKI obtains 35% MAP improvement over the best performing PCNN baseline. In contrast to baseline models, our approach leverages the neighborhood information of each entity from the text predicates in the 2007 corpus and predicates / relations from the 2005-2006 corpus. This background knowledge contributes to the significant performance improvement.

⁶This data can be downloaded from <http://iesl.cs.umass.edu/riedel/ecml/>

⁷Facts of ‘NA’ (no relation) in the test data are not included in the evaluation process.

Note that, our model uses only the graph information from the entity neighborhood and does not use any text encoder such as Piecewise Convolutional Neural Nets (PCNN) (Zeng et al., 2015), where convolutional neural networks were applied with piecewise max pooling to encode textual sentences. This further demonstrates the importance of entity neighborhood information for relation inference. It is possible to further improve the performance of our model by incorporating text encoders as an additional signal. Some prior works (Verga et al., 2016; Toutanova et al., 2015) also leverage text encoders for relation inference.

5 Related Work

Relation Extraction: Mintz et al. (2009) utilize the entity pair overlap between knowledge bases and text corpus to generate signals for automatic supervision. To avoid false positives during training, many works follow the at-least-one assumption, where at least one of the text patterns between the entity pair indicate an aligned predicate in the KB (Hoffmann et al., 2011; Surdeanu et al., 2012; Zeng et al., 2015; Lin et al., 2016). These works do not leverage graph information. In addition, Universal Schema (Riedel et al., 2013; Verga et al., 2017) tackled this task by low-rank matrix factorization. Toutanova et al. (2015) exploit graph information for knowledge base completion. However, their work cannot deal with unseen entities since entities' parameters are explicitly learned during training.

Schema Mapping: Traditional schema mapping methods (Rahm and Bernstein, 2001) involve three kinds of features, namely, language (name or description), type constraint, and instance level co-occurrence information. These methods usually involve hand-crafted features. In contrast, our model learns all the features automatically from OpenIE and KB with no feature engineering. This makes it easy to scale to different domains with little model tuning. Also, the entity types used in traditional schema mapping is always pre-defined and coarse grained, so cannot provide precise constraint of relations for each entity. Instead, our ENE model automatically learns soft and fine-grained constraints on which relations entities are likely to participate in. It is also compatible with pre-defined type systems.

Relation Grounding from OpenIE to KB: Instead of modeling existing schema, open informa-

tion extraction (OpenIE) (Banko et al., 2007; Yates et al., 2007; Fader et al., 2011; Mausam et al., 2012) regards surface text mentions between entity pairs as separate relations, and do not require entity resolution or linking to KB. Since they do not model KB, it is difficult to infer KB relations only based on textual observations. Soderland et al. (2013) designed manual rules to map relational triples to slot types. Angeli et al. (2015) used PMI between OpenIE predicates and KB relations using distant-supervision from shared entity pairs for relation grounding. Yu et al. (2017) used word embedding to assign KB relation labels to OpenIE text predicates without entity alignment. These works do not exploit any graph information.

Entity Modeling for Relation Grounding: People leveraged several entity information to help relation extraction. Zhou et al. (2005) employed type information and observed 8% improvement of F-1 scores. Ji et al. (2017) encoded entity description to calculate attention weights among different text predicates within an entity pair. However, entity type and description information is not commonly available. Instead, the neighborhood information is easier to obtain and can also be regarded as entities' background knowledge. Universal Schema (Riedel et al., 2013) proposed an E-Model to capture entity type information. However, it can easily overfit in the OpenIE setting with large number of entities and a sparse knowledge graph.

6 Conclusion

In this work we jointly leverage relation mentions from OpenIE extractions and knowledge bases (KB) for relation inference and aligning OpenIE extractions to KB. Our model leverages the rich information (KB relations and OpenIE predicates) from the neighborhood of entities to improve the performance of relation inference. This also allows us to deal with new entities without using any entity-specific parameters. We further explore several attention mechanisms to better capture entity pair information. Our experiments over several datasets show 33.5% MAP improvement on average over state-of-the-art baselines.

Some future extensions include exploring more advanced graph embedding techniques without modeling entity-specific parameters and using text encoders as additional signals.

References

- 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 (Volume 1: Long Papers)*, volume 1, pages 344–354.
- Michele Banko, Michael J Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *IJCAI*, volume 7, pages 2670–2676.
- Mirko Bronzi, Valter Crescenzi, Paolo Merialdo, and Paolo Papotti. 2013. Extraction and integration of partially overlapping web sources. *Proceedings of the VLDB Endowment*, 6(10):805–816.
- Michael J Cafarella, Alon Halevy, Daisy Zhe Wang, Eugene Wu, and Yang Zhang. 2008. Webtables: exploring the power of tables on the web. *Proceedings of the VLDB Endowment*, 1(1):538–549.
- Mark Dredze, Paul McNamee, Delip Rao, Adam Gerber, and Tim Finin. 2010. Entity disambiguation for knowledge base population. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 277–285. Association for Computational Linguistics.
- 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 2011, 27-31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1535–1545.
- Qiang Hao, Rui Cai, Yanwei Pang, and Lei Zhang. 2011. From one tree to a forest: a unified solution for structured web data extraction. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 775–784. ACM.
- Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 541–550. Association for Computational Linguistics.
- Guoliang Ji, Kang Liu, Shizhu He, Jun Zhao, et al. 2017. Distant supervision for relation extraction with sentence-level attention and entity descriptions. In *AAAI*, pages 3060–3066.
- Heng Ji, Joel Nothman, Ben Hachey, et al. 2014. Overview of tac-kbp2014 entity discovery and linking tasks. In *Proc. Text Analysis Conference (TAC2014)*, pages 1333–1339.
- Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Thomas Lin, Oren Etzioni, et al. 2012. Entity linking at web scale. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*, pages 84–88. Association for Computational Linguistics.
- Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 2124–2133.
- Colin Lockard, Prashant Shiralkar, and Xin Luna Dong. 2019. When open information extraction meets the semi-structured web. In *NAACL-HLT*. Association for Computational Linguistics.
- 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 2012, July 12-14, 2012, Jeju Island, Korea*, pages 523–534.
- Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.
- Dan Olteanu, Holger Meuss, Tim Furche, and François Bry. 2002. Xpath: Looking forward. In *XML-Based Data Management and Multimedia Engineering - EDBT 2002 Workshops, EDBT 2002 Workshops XMLDM, MDDE, and YRWS, Prague, Czech Republic, March 24-28, 2002, Revised Papers*, pages 109–127.
- Pengda Qin, Weiran Xu, and William Yang Wang. 2018. Robust distant supervision relation extraction via deep reinforcement learning. *arXiv preprint arXiv:1805.09927*.
- Erhard Rahm and Philip A Bernstein. 2001. A survey of approaches to automatic schema matching. *the VLDB Journal*, 10(4):334–350.
- Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer.
- Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M Marlin. 2013. Relation extraction with

matrix factorization and universal schemas. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 74–84.

Stephen Soderland, John Gilmer, Robert Bart, Oren Etzioni, and Daniel S Weld. 2013. Open information extraction to kbp relations in 3 hours. In *Proc. Text Analysis Conference (TAC2013)*.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 455–465. Association for Computational Linguistics.

Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoi-fung Poon, Pallavi Choudhury, and Michael Gamon. 2015. Representing text for joint embedding of text and knowledge bases. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Patrick Verga, David Belanger, Emma Strubell, Benjamin Roth, and Andrew McCallum. 2016. [Multilingual relation extraction using compositional universal schema](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 886–896, San Diego, California. Association for Computational Linguistics.

Patrick Verga, Arvind Neelakantan, and Andrew McCallum. 2017. [Generalizing to unseen entities and entity pairs with row-less universal schema](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 613–622, Valencia, Spain. Association for Computational Linguistics.

Alexander Yates, Michele Banko, Matthew Broadhead, Michael J. Cafarella, Oren Etzioni, and Stephen Soderland. 2007. Texrunner: Open information extraction on the web. In *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, April 22-27, 2007, Rochester, New York, USA*, pages 25–26.

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 (Volume 1: Long Papers)*, volume 1, pages 854–864.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762.

Guodong Zhou, Jian Su, Jie Zhang, and Min Zhang. 2005. Exploring various knowledge in relation extraction. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 427–434. Association for Computational Linguistics.

A Appendices

A.1 Derivation of Bayesian Inference Baselines

$$\begin{aligned} P(p|\bar{p}') &= \max_{p' \in R(s,o)} P(p|p') = \max_{p' \in R(s,o)} \frac{P(p,p')}{P(p')} \\ &= \max_{p' \in R(s,o)} \frac{\#(p,p')/\#entity\ pair}{\#p'/\#entity\ pair} \\ &\approx \max_{p' \in R(s,o)} \frac{\#(p,p') + \Delta}{\#(p') + |R^{text} \cup R^{KB}| \Delta} \end{aligned}$$

$R(s,o)$ is the set of observed predicates between subject s and object o . Δ is a smoothing factor which we choose $1e-6$ in our implementation. Using conditional independence assumptions (refer to Figure 2):

$$\begin{aligned} P(p|s,o) &= \frac{P(p,s,o)}{P(s,o)} = \frac{P(p|s)P(s)P(o|p)P(o)}{P(s)P(o)} \\ &= \sum_{p_s^N \in R(s,\cdot)} P(p_s^N|s)P(p|p_s^N) \\ &\cdot \sum_{p_o^N \in R(\cdot,o)} P(p_o^N|o)P(p|p_o^N) \end{aligned}$$

$$\begin{aligned} P(p|s,\bar{p}',o) &= \frac{P(p,s,\bar{p}',o)}{P(s,\bar{p}',o)} \\ &= \frac{P(p|s)P(s)P(p|\bar{p}')P(\bar{p}')P(o)P(o)}{P(s)P(\bar{p}')P(o)} \\ &= P(p|\bar{p}') \sum_{p_s^N \in R(s,\cdot)} P(p_s^N|s)P(p|p_s^N) \\ &\cdot \sum_{p_o^N \in R(\cdot,o)} P(p_o^N|o)P(p|p_o^N) \end{aligned}$$