ReMine: Open Information Extraction with Global Information

Qi Zhu
University of Illinois
Urbana-Champaign, Illinois
qiz3@illinois.edu

Xiang Ren
University of Southern California
Los Angles, California
xiangren@usc.edu

Jingbo Shang
University of Illinois
Urbana-Champaign, Illinois
shang7@illinois.edu

Yu Zhang
University of Illinois
Urbana-Champaign, Illinois
yuz9@illinois.edu

Frank F. Xu
Shanghai Jiao Tong University
Shanghai, China
frankxu@sjtu.edu.cn

Jiawei Han
University of Illinois
Urbana-Champaign, Illinois
hanj@illinois.edu

ABSTRACT

Extracting entities and their relations from text is an important task for understanding massive text corpora. Open information extraction (IE) systems mine relation tuples (i.e., entity arguments and a predicate string to describe their relation) from sentences, and do not confine to a pre-defined schema for the relations of interests. However, current open IE systems focus on modeling *local* context information in a sentence to extract relation tuples, while ignoring the fact that *global* statistics in a large corpus can be *collectively* leveraged to identify high-quality sentence-level extractions. In this paper, we propose a novel open IE system, called ReMine, which integrates local context signal and global structural signal in a unified framework with distant supervision. The new system can be efficiently applied to different domains as it uses facts from external knowledge bases as supervision; and can effectively score sentence-level tuple extractions based on corpus-level statistics. Specifically, we design a joint optimization problem to unify (1) segmenting entity/relation phrases in individual sentences based on local context; and (2) measuring the quality of sentence-level extractions with a translating-based objective. Experiments on two real-world corpora from different domains demonstrate the effectiveness and robustness of ReMine when compared to other open IE systems.

1 INTRODUCTION

Massive corpora are emerging worldwide in different domains and languages. The sheer size of such data and the fast pace of new data generation make manual curation unscalable and infeasible. Information extraction (IE), i.e., entity and relation extraction, is a key step towards automated knowledge acquisition. It has a large variety of downstream applications. For example, people can further build a knowledge base upon that and turn information into structured storage. Question answering systems also require such extracted entities and relations to provide accurate responses to human queries.

There are numerous efforts trying to turn raw corpus into structured knowledge representation, *e.g.* named entity recognition, relation extraction, semantic role labeling, *etc.* Almost all the state-of-the-art methods involve human experts or build upon carefully human annotated benchmark datasets. These domain-dependent methods can easily suffer from cross-domain adaptation and subtle linguistic difference among languages. Recently, researchers proposed several open-domain IE systems [2, 6, 7, 13, 16] that do not require much human curation. Two major challenges of such systems are (1) how to avoid complicated human annotations; and (2) how to improve the reliability on the extracted information. To address the first concern, previous studies mainly rely on reliable patterns created by linguistic experts, introducing weak supervision [4, 9], or so-called distant supervision [12, 14] from the external knowledge base. Open IEs distinguish informative tuples at the sentence-level and only acquire limited user-provided seeds [7, 16] or distant training corpus [2]. Existing Open IEs all obtain information from local context, *e.g.* POS tags, dependency tree, *etc.* but do not benefit from scale of the corpus. Phrase mining methods [10, 17] can generate high-quality phrases at the corpus level, which further incorporates global statistical features but they cannot appeal well-organized extractions.

In this paper, we study open IE problem from a unified perspective for massive text corpora, as shown in Fig. 1. First, ReMine will identify entity and relation phrases from local context. For example, suppose we have a sentence "Your dry cleaner set out from eastern Queens on foot Tuesday morning and now somewhere near Maspeth.". We will first extract three entity phrases, *eastern Queens*, *Tuesday morning*, *Maspeth*, as well as two background phrases *Your dry cleaner*, *foot*. Then, ReMine jointly mines relation tuples and measure extraction with global translating objective. Local consistent text segmentation may generate noisy tuples, such as <your dry cleaner, *set out from*, eastern Queens> and <eastern Queens, *on*, foot>. However, from the global cohesiveness view, we may infer the second tuple as a false positive. Entity phrases like "eastern Queens" are seldom linked by relation phrase *"on"* in extracted tuples. Overall, ReMine will iteratively refine extracted tuples and learn entity and relation representation at corpus level.

With careful attention to advantages of linguistic patterns [7, 8] and representation learning in knowledge base [3], this approach benefits from both side. Compared to previous open IE systems, ReMine improve extracted tuples via global cohesiveness and its accuracy is not sensitive to the target domain.

The major contributions of this paper can be summarized as follows.

(1) We propose a novel open IE framework, ReMine, that can mine information tuples with local information and global statistics.

Conference'17, July 2017, Washington, DC, USA
2017. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

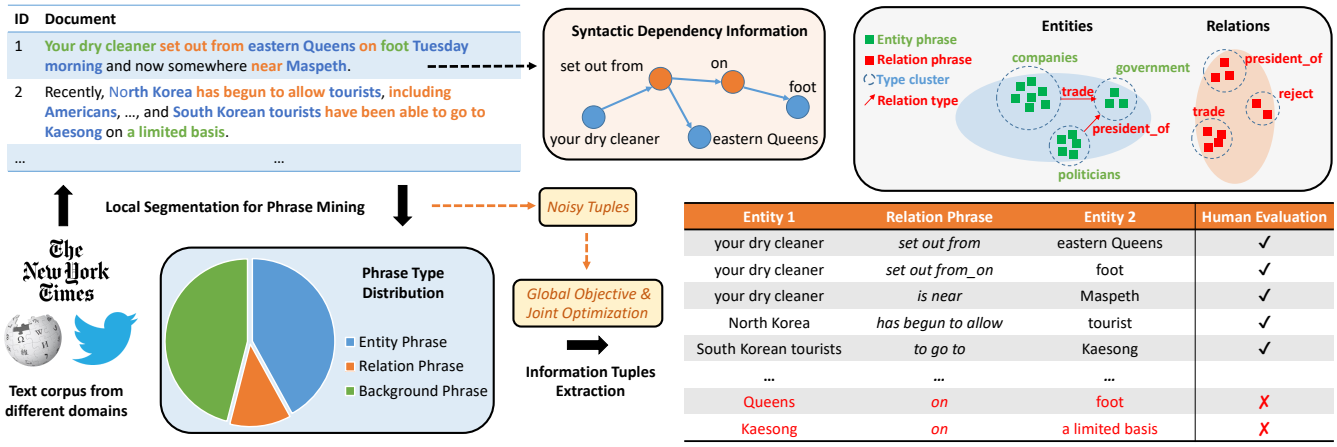


Figure 1: The Overview of the ReMine Framework.

(2) We develop a context-agnostic phrasal segmentation algorithm can identify high quality phrases of multiple types.
(3) Extensive experiments on three public datasets demonstrate that ReMine achieves higher precision and recall on several IE tasks.

2 BACKGROUND AND PROBLEM DEFINITION

Generally, Open IE can be defined as a two-stage extraction tasks, first identify entity phrases $\mathcal{E}$, relation phrases $\mathcal{R}$. By selecting and pairing up entity phrases into entity arguments, it further extract meaningful relation tuples $\mathcal{T}$ among them.

Definition 2.1. **(Entity Phrases)** *For any sentence s in a corpus $\mathcal{D}$, entity phrases $(e_1, e_2, ..., e_n) \subset \mathcal{E}$ are defined as token spans in s, which could be different types* e.g. *time, location, country, organization.*

In Open IE, entity phrases are usually not only name entities. For example, in sentence "Your dry cleaner set out from eastern Queens on foot Tuesday morning and is now somewhere near Maspeth.", "your dry cleaner" is not a name entity, however, it is the subject of this sentence and can not be omitted in relation tuples extraction.

Definition 2.2. **(Relation Phrases)** *Given entity phrases $(e_1, e_2, ..., e_n)$ in sentence s, relation phrases $(r_1, r_2, ..., r_n) \subset \mathcal{R}$ are defined between a specific entity pair (e_a, e_b). For each relation phrase r, it conveys some semantic correlations between entity phrases.*

Relation phrases are not exactly relation types, specifically, one relation type can correspond to multiple relation phrases, ***i.e.*** **location/country/capital can correspond to ('s capital, capital of, the capital, ...)**.

Definition 2.3. **(Background Phrases)** *Besides entity and relation phrases, there are phrases that correspond to less known facts,* **e.g. *the major effort, the final group***, *which contains information in specific sentence whereas less import in corpus.*

Definition 2.4. **(Information Tuple)** *For any sentence s in a corpus $\mathcal{D}$, information tuples $\mathcal{T}$ are defined as $(h,(l_1, l_2, ..., l_n),t)$, where h and t correspond to head and tail entity arguments and set* **l** *in correspondence to predicate.*

Different from existing Open Information Extraction systems [2, 16], we treat predicate as combination of relation phrase, *e.g.* "Your dry cleaner *set_out_from_[entity] on* foot". Unlike knowledge base, predicate in tuples cannot simply impressed by one clear relation type. So we will show how we capture semantic drift in one predicate and identify relation phrases $(l_1, l_2, ...l_n)$ in our formulation.

3 THE REMINE FRAMEWORK

ReMine aims to jointly address two sub problems, that is, extracting entity & relation phrases and mining relation tuples. There are two challenges respectively, first, distant supervision may contain "false" entities and relation seeds, robust quality score need to be assigned on every phrase. Second, there exists multiple entity phrases in one sentence, selecting head and tail entity arguments for relation tuples may suffer from local structure ambiguity.

Framework Overview. We proposed a framework that integrate both local context and global statistics called ReMine (see also Fig. 1) as follows:

(1) Do context-agnostic phrasal segmentation on target corpus, to generate entity phrases $\mathcal{E}$ and relation phrases $\mathcal{R}$. Apply random forest to obtain phrase type and quality from partially labeled training data $\mathcal{D}_L$ with distant supervision.
(2) Identify predicate l between entity argument pair and organize sentence-level relation tuples $\mathcal{T}$ based on local segmentation objective.
(3) Learn entity and relation representations $\mathcal{V}$ via global translating objective.
(4) Update sentence-level extractions with joint information from local context and global statistics, reduce local error by checking its global cohesiveness.

Table 1: Entity and Relation Phrases Regular Patterns, where V=<VB|VBD|VBG|VBN|VBN|VBP|VBZ>+, W=<IN|RP>?, P=<NN|JJ|RP|PRP|DT>

Pattern	Examples
<DT|PP$>?<JJ>*<NN>+	the state health department
<NNP>+<IN>?<NNP>+	Gov. Tim Pawlenty of Minnesota
V	furnish, work, leave
VP	provided by, retire from
VW^*P	die peacefully at home in

3.1 Local Segmentation for Entity and Relation Mining

We explore entity and relation mining as a multiple type phrasal segmentation task, traditional Open IE use NP-chunking to extract entity phrases, yet not all noun phrases can carry rich information and it requires additional training. Our method uses context-agnostic phrasal segmentation to detect and evaluate whether a token span more likely to be an entity phrase, relation phrase or background phrase. Given word sequence C and corresponding linguistic features $\mathcal{F}$, a segmentation $\mathcal{S} = s_1, s_2, ..., s_n$ is separated by boundary index $B = b_1, b_2, ..., b_{n+1}$. For each segment s_i, there is a type $t_i \in \{entity, relation, background\}$, indicating the most possible type of s_i, the joint probability is factorized as:

$$P(\mathcal{S}, C, \mathcal{F}) = \prod_{t=1}^{n} P(b_{t+1}, w_{[b_t, b_{t+1})} | b_t, \mathcal{F}) \tag{1}$$

ReMine generates each segment as follows,
1. Given start index b_i, generate end index b_{i+1} according to context-agnostic prior Δ, *i.e.* dependency tree pattern prior.

$$P(b_{i+1} | b_i, \mathcal{F}) = \Delta(\mathcal{F}_{[b_i, b_{i+1})}]) \tag{2}$$

2. Given the start and end index (b_i, b_{i+1}) of segment s_i, generate word sequence $w_{[b_i, b_{i+1})}$ according to a multinomial distribution over all segments at the same length.

$$P(w_{[b_i, b_{i+1})} | b_i, b_{i+1}) = P(w_{[b_i, b_{i+1})} | b_{i+1} - b_i) \tag{3}$$

3. Finally, we generate a phrase type t_i indicating that category $w_{[b_i, b_{i+1})}$ most likely belongs and a quality score showing how it likely to be a good phrase.

$$P(\lceil w_{[b_i, b_{i+1})} \rfloor | w_{[b_i, b_{i+1})}) = \max_{t_i} P(t_i | w_{[b_i, b_{i+1})}) \tag{4}$$

3.1.1 Candidate Generation. Phrase Mining had made an assumption that quality phrases are frequent n-grams in corpus, while it is not the case when sentence level extractions are important. To overcome phrase sparsity, we adopt several NP Chunking rules, see Table 1, are adopted to discover infrequent but informative phrase candidates. In our experiments, frequent n-grams and NP chunking rules contribute comparable amount of phrase candidates.

3.1.2 Phrase Features. In order to estimate type & quality, we designed a set of features $\mathcal{F}$ in Table 2 that indicates a good phrase and its type. It can be grouped into several different categories, *i.e.* statistic features, token-wise features, POS features.

Algorithm 1: Viterbi Training

Input: Corpus $\mathcal{D}$ and phrase quality Q
Output: Δ and θ_u

```
initialization;
while θ_u does not converge do
    while Δ does not converge do
        B ⇐ best segmentation via Eq. 5;
        update Δ using B according to Eq. 6;
    end
    B ⇐ best segmentation via Eq. 5;
    update θ_u using B according to Eq. 7;
end
return Δ and θ_u
```

3.1.3 Robust Distant Training. We notice that both distant training phrases and NP chunking rules are noisy. To approach noisy distant training data, previous phrasal segmentation work [10, 17] proposed Robust Positive-Only Distant Training based on random forest. We follow this line of work, construct positive and negative pool from distant linking data and phrase candidates. Given the theoretic bound of ensemble learning, when amount of decision trees are large enough, the misclassification ratio of the unlabeled phrases can be as low as expected.

We denote $P(b_{i+1} | b_i, \mathcal{F})$ as $\Delta\mathcal{F}_{[b_i, b_{i+1})}]$, $P(w_{[b_i, b_{i+1})} | b_{i+1} - b_i)$ as θ_u and $\max_{t_i} P(t_i | w_{[b_i, b_{i+1})})$ as $Q(w_{[b_i, b_{i+1})})$. Now we will show how we use Viterbi Training [1] to update Segmentation S and parameters θ, Δ iteratively. In the E-step, given θ and Δ, dynamic programming is used to find the optimized segmentation. Given start index i and end index j,

$$\mathbf{H}_j = \max(\mathbf{H}_j, \mathbf{H}_i \cdot p(i, w_{[i,j)} | j, \mathcal{F})) \tag{5}$$

where $\mathbf{H}_i$ the current maximum generation probability ends at i. In the M-step, we first fixed parameter θ, and update context-agnostic prior, $f_{dep} \in \mathbb{N}$ denotes tree pattern id:

$$\Delta(f_{dep}) = \frac{\sum_{i=1}^{m} \mathbf{1} \cdot (\mathcal{F}_{[b_i, b_{i+1})} = f_{dep})}{\sum_{l=2}^{max_l} \sum_{i=1}^{n-l} \mathcal{F}_{[i, i+l)]}} \tag{6}$$

Next when Δ is fixed, optimized solution of θ_u is:

$$\theta_u = \frac{\sum_{i=1}^{m} \mathbf{1} \cdot (w_{[b_i, b_{i+1})} = u)}{\sum_{i=1}^{m} \mathbf{1} \cdot (b_{i+1} - b_i = |u|)} \tag{7}$$

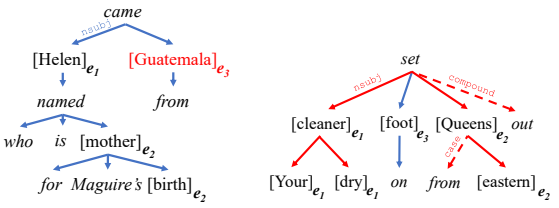
3.2 Intergrated Relation Tuple Extraction

In last section, we have introduced how ReMine extract entity/relation phrases using local segmentation and global cohesiveness. Now we will first demonstrate how we extract relation tuples based on local segmentation.

Definition 3.1. **(Semantic Path)** *For any given two entity arguments (head, tail) in the same sentence, semantic path is defined as selected dependency path $P_{h,t} = \{p_1, p_2, ..., p_n\}$ between token span* **h** *and* **t**.

Table 2: Feature List

Feature	Descriptions	Example
popularity	raw frequency, occurrence probability, log occurrence probability	
completeness	sub-phrases within long frequent phrases are also informative	*"relational database system" meets the criteria*
concordance	tokens in quality phrases should co-occurs frequently	*"strong tea" versus "heavy tea"*
punctuations	phrase in parenthesis, quote or has dash after	*(12.pm), "the Zeitlin sidewinder"*
stopwords	first/last token is stopword and stopword ratio	*the, their, therefore*
word shape	first capitalized or all capitalized	*NBA, Defense Secretary Donald H. Rumsfeld*
part-of-speech tags	unigram and bigram POS tags	*the Rev. Ian Paisley DT,NNP,DT-NNP*



(a) Selecting among candidate subject entities **(b) Finding shortest dependency path**

Figure 2: Dependency parse tree of example sentence in Fig. 1, *"Your dry cleaner set out from eastern Queens on foot Tuesday morning."* Segmented entities are marked as "[entity_token]$_{e_i}$"

3.2.1 Positive entity pairs generation. For a given sentence s, with local segmentation, we have entity arguments $e_1, e_2, ..., e_n$ and relation arguments $r_1, r_2, ..., r_n$, notice that good background phrases also recognized as arguments. However, it's infeasible to explore semantic path between every entity pair and a large portion of tuples are incorrect among $N(N-1)$ pairs. Positive entity pairs E_p^+ are entity arguments pair selected. Here we heuristically initialize E_p^{+0} by attaching *nearest* subject e_i to object e_j and make an approximation that each entity argument phrase can only be object once. Nearest subject of e_j is defined as entity e_i that has shortest dependency path length to e_j among all other entities. Considering Fig. 2a, we would like find subject of entity e_3 : *Guatemala*, length of shortest path between e_3 and e_1, e_2 are 2,4 respectively. For those entity candidates with the same distance, see Fig. 2b, both e_1: *Your dry cleaner* and e_2: *eastern Queens* is one hop away from e_2: *foot*. We will prefer subject with "nsubj" type *i.e.* e_1 then choose closest entity in original sentence if there are still multiple of them.

3.2.2 Local Consistency Objective.

$$O_{local} = \sum_{(e_i,e_j)\in E_p^+} \log P(\mathcal{S}, P_{(e_i,e_j)}, \mathcal{F}) \tag{8}$$

Once E_p^+ is determined, the semantic path is therefore along the shortest dependency path between two arguments. For example, in Fig. 2b, the semantic path between e_1 and e_4 is marked in red. To preserve integrity of potential relation phrases, particles and preposition along semantic path are added as red dotted line in Fig. 2b. To find relation phrases on semantic path, we adopt phrasal segmentation objective mentioned before as shown in Eq. 8. Leveraging information along the dependency path between two given entities have been proved useful for closed-domain relation classification and extraction, open information extraction, as well as event extraction tasks as it reduces noise by removing irrelevant semantic phrases or clauses in long sentences with multiple entities. In our work, we assume that the information along the proposed semantic path is sufficient for relation phrase mining.

3.2.3 Global Objective. To model the global cohesiveness of extracted relation tuples, we adopt translating measuring of knowledge base completion [3]. With global measuring of relation tuples, we have global objective to associate extracted relation tuples in the corpus $\mathcal{D}$ as below,

$$O_{global} = -\sum_{(h,l,t)\in E^+_{h,l,t}} \sum_{(h',l,t')\in E^-_{h',l,t'}} \gamma + \|h+l-t\| - \|h'+l-t'\| \tag{9}$$

where $E^+_{h,l,t}$ denote for $(h,t) \in E_p^+$ and predicate l stands for average extracted semantic path $l = (r_1, r_2, ..., r_n)$ in between, γ is the hyper margin, $E^-_{h',l,t'}$ is composed of training tuples with either $\mathbf{h}$ or $\mathbf{t}$ replaced. We use L_1 norm in ReMine for efficiency. For each tuple (h,l,t) in local optimization stage, especially, for relation phrases $\mathcal{R}$, we learn their semantic representation as $l = \bar{\mathcal{R}}$. Global objective tries to minimize dissimilarities for positive extractions, which start with current positive extractions and iteratively propagate to more unknown tuples in local optimization.

3.2.4 Update Positive Pairs.

$$E_p^+ = \underset{x\in e_h}{\operatorname{argmin}} \|x + \mathbf{P}(x, e_t) - e_t\| \tag{10}$$

Given semantic representation for each entity $\mathbf{e}$ and relation $\mathbf{r}$ and local segmentation between entity pairs. We can, therefore, update the Positive Entity Pairs by finding most semantically consistent subject $\mathbf{e}_h$ for each object $\mathbf{e}_t$. This task is identical to link prediction in the literature.

3.2.5 The Integrated Optimization Problem.

$$O = O_{local} + O_{global} \tag{11}$$

To maximize above unified open IE objective, see Alg. 2, we first initialize positive entity pairs E_p^{+0}. Given entity argument pairs, we perform local segmentation on its semantic path, which leads to information tuples. Note that, at the first round, there are no global representation, so we preserve every relation phrase in the semantic path. Then we update global phrase semantic representation via

Algorithm 2: Joint Tuple Mining

Input: Corpus $\mathcal{D}$, Sentence S, Entities $\mathcal{E}$, Relations $\mathcal{R}$
Output: Relation Tuples $\mathcal{T}$, representation $\mathcal{V}$

initialize positive E_p^+ according Sec. 3.2.1 ;
while L *does not convergence* **do**
  for *each entity argument pair* (e_i, e_j) **do**
    identify semantic path $P(e_i, e_j)$;
    $l \Leftarrow$ find best semantic path segmentation via local objective;
    construct relation tuple as (e_i, l, e_j) ;
  end
  $\mathcal{V} \Leftarrow$ learn representation for relation tuples via global objective;
  update E_p^+ according to $\mathcal{E}$, $\mathcal{R}$ and $\mathcal{V}$;
end

global objective. With both global semantic information and local segmentation result, ReMine updates Positive Pairs as described in Sec. 3.2.1. We iteratively updating local and global objective until it convergence, which will lead to a stable positive entity pairs. In experiments, we discover empirically, iteration can be stopped after second round.

4 EXPERIMENTS

4.1 Experimental Setup

4.1.1 Datasets. We use three datasets in our experiments: (1) NYT [15]: The training corpus consists of 236k sentences from ~294k 1987-2007 New York Times news articles. 395 sentences are manually annotated with entity and relation mentions by authors [15]. (2) Twitter [18]: The dataset consists of 1.4 million tweets in Los Angeles with entities and/or noun phrases collected from 2014.08.01 to 2014.11.30.

4.1.2 Training Corpora Distant Linking. Our proposed method ReMine mainly have several outcomes, including high quality entity/relation phrases and information tuples. For each corpus, we first generates some distant supervision seeds via DBPedia Spotlight service[1] [5] for entity phrases. With entity phrases, we generates relation phrases between each pair of entity mentions via pattern matching. We then followed the procedure introduced in Sec. 3.1, segmenting input corpora into entity phrases, relation phrases and background phrases.

4.1.3 Feature Generation. We described all text features used in phrase extraction and tuple mining Table. 2. Our methods followed [17] to generate context-free features for phrase candidates. We applied the Stanford CoreNLP [11] tool to get POS tags and dependency parsing tree. We use same external linguistic features as other Open IE methods in our experiments.

4.1.4 Compared Methods. We compare ReMine with the following state-of-the-art information extraction methods including both pattern-based and clause-based methods: (1) OLLIE [16]: utilizes open pattern learning and extracts patterns over dependency path and part-of-speech tags. (2) ClausIE [6] adopts clause patterns to handle long distance relationships. (3) Stanford OpenIE [2] learns a clause splitter via distant training data. All Open IE methods, to some extent, requires weak supervision or distant supervision.

4.1.5 Evaluation Metrics. For the Information Tuple Discovery task, since each tuple obtained by ReMine and other benchmark methods will also be assigned a confidence score. We rank all the tuples according to their confidence scores. Based on the ranking list, we use the following four measures: *P@k* is the precision at rank *k*. *MAP* is mean average precision of the whole ranking list. *NDCG@k* is the normalized discounted cumulative gain at rank k. *MRR* is the mean reciprocal rank of the whole ranking list. Note that we do not use recall in this task because it is infeasible to know all the "correct" tuples.

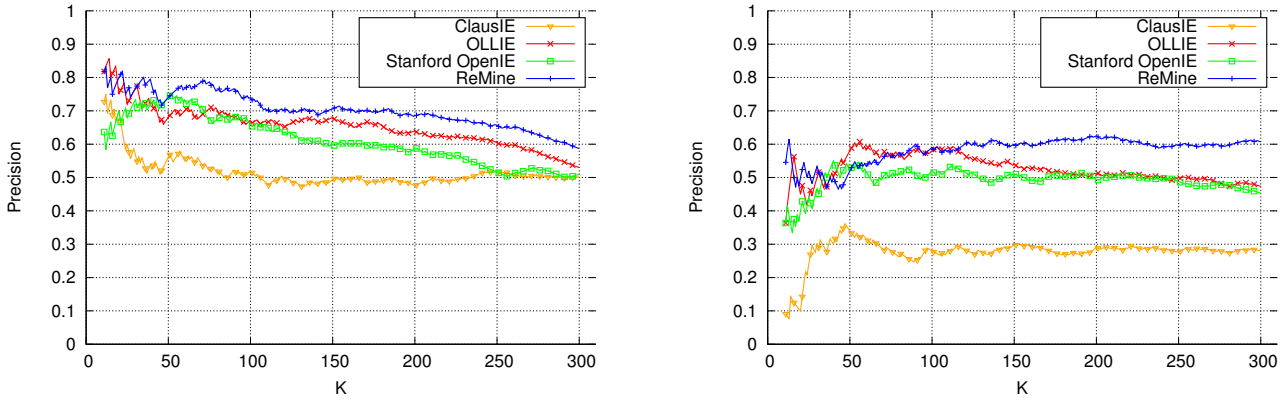
4.2 Experiments and Performance Study

Open IE system can extract information tuples from open domain corpus. We compared ReMine with three Open IE systems mentioned above. We manually labeled the extractions got from ReMine and other three baseline extractors. Each extraction was labeled by two independent annotators for 2 rounds. Both annotators are highly proficient and literate in English. The two annotators are asked to evaluate without knowing which model produced the results, eliminating potential bias in evaluation. In the second round, extractions will be relabeled by the two labelers if they have different opinions during the first round. Similar to the settings in previous study [6], we ignore the context of the extracted tuples during labeling.

Among all the Open IE system described above, ReMine and OLLIE extract a relatively small number of tuples. For example, for the first 100 sentences in the NYT test set, both ReMine and OLLIE get about 300 tuples. In contrast, Stanford OpenIE returns more than 1,000 tuples. It may be unfair if we directly plot the *P@k* curves to compare those methods and ignore the tuple numbers. For example, imagine there is a system returning N tuples. It is not difficult to paraphrase each of them and get another N tuples. If we use the whole $2N$ tuples to plot a *P@k* curve, we are essentially "stretching" the original curve to a longer one. Since *P@k* curves are usually monotone decreasing, we will have a "higher" curve after the paraphrase. Due to this problem, we randomly sample 300 tuples for each Open IE system to plot the curves. The results are shown in Figure 3 and Table 3.

According to the curves in Figure 3a and 3b, ReMine achieves the best performance among all Open IE systems. In the NYT dataset, OLLIE, Stanford OpenIE and ClausIE actually have similar overall precision (*i.e.* *P*@300). But OLLIE has a "higher" curve since most tuples obtained by Stanford OpenIE and ClausIE will be assigned score 1. Therefore we can only randomly shuffle these score-1 tuples and may not rank them in a very rational way. In contrast, the scores of different tuples obtained by OLLIE and ReMine are usually distinct from each other. In Table 3, ReMine also consistently performs the best according to the rank-based measures. In the Twitter dataset, ClausIE has a rather low score since there are lots of non-standard language usages and grammatic errors in tweets.

[1] https://github.com/dbpedia-spotlight/dbpedia-spotlight



(a) NYT **(b) Twitter**

Figure 3: Precision@K on both datasets

Table 3: Performance of different methods.

Methods	NYT						Twitter					
	P@100	P@200	MAP	NDCG@100	NDCG@200	MRR	P@100	P@200	MAP	NDCG@100	NDCG@200	MRR
ClausIE	0.520	0.475	0.534	0.567	0.632	0.027	0.280	0.290	0.292	0.309	0.522	**0.024**
Stanford OpenIE	0.660	0.585	0.630	0.655	0.726	0.023	0.520	0.495	0.493	0.468	0.620	0.016
OLLIE	0.670	0.640	0.683	0.684	0.775	**0.028**	0.580	0.510	0.525	0.519	0.626	0.017
ReMine	**0.740**	**0.685**	**0.726**	**0.757**	**0.776**	0.027	**0.590**	**0.625**	**0.593**	**0.585**	**0.657**	0.022

Therefore clause-based methods may not achieve a satisfying performance. In contrast, ReMine shows its power in dealing with short and noisy text.

5 CONCLUSION

This paper studies the task of open information extraction and proposes a principled framework, ReMine, to unify local contextual information and global structural cohesiveness for effective extraction of relation tuples. ReMine leverages distant supervision in conjunction with existing knowledge bases to provide automatically-labeled sentence and guide the entity and relation segmentation. The local objective is further learned together with a translating-based objective to enforce structural cohesiveness, such that corpus-level statistics are incorporated for boosting high-quality tuples extracted from individual sentences. We develop a joint optimization algorithm to efficiently solve the proposed unified objective function and can output quality extractions by taking into account both local and global information. Experiments on two real-world corpora of different domains demonstrate that ReMine system achieves superior precision when outputting same number of extractions, compared with several state-of-the-art open IE systems.

REFERENCES

[1] Armen Allahverdyan and Aram Galstyan. 2011. Comparative analysis of viterbi training and maximum likelihood estimation for hmms. In *NIPS'11*. MIT Press, 1674–1682.

[2] Gabor Angeli, Melvin Johnson Premkumar, and Christopher D Manning. 2015. Leveraging linguistic structure for open domain information extraction. In *ACL'15*. ACL, 344–354.

[3] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *NIPS'13*. MIT Press, 2787–2795.

[4] Razvan C. Bunescu and Raymond J. Mooney. 2007. Learning to Extract Relations from the Web using Minimal Supervision. In *ACL'07*.

[5] Joachim Daiber, Max Jakob, Chris Hokamp, and Pablo N. Mendes. 2013. Improving Efficiency and Accuracy in Multilingual Entity Extraction. In *Proceedings of the 9th International Conference on Semantic Systems (I-Semantics)*.

[6] Luciano Del Corro and Rainer Gemulla. 2013. Clausie: clause-based open information extraction. In *WWW'13*. ACM, 355–366.

[7] Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *EMNLP'11*. ACL, 1535–1545.

[8] Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *ACL'92*. ACL, 539–545.

[9] Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke S. Zettlemoyer, and Daniel S. Weld. 2011. Knowledge-Based Weak Supervision for Information Extraction of Overlapping Relations. In *ACL'11*.

[10] Jialu Liu, Jingbo Shang, Chi Wang, Xiang Ren, and Jiawei Han. 2015. Mining quality phrases from massive text corpora. In *SIGMOD'15*. ACM, 1729–1744.

[11] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit.. In *ACL'14 (System Demonstrations)*. ACL, 55–60.

[12] Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *ACL-IJCNLP'09*.

[13] Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 2012. PATTY: A taxonomy of relational patterns with semantic types. In *EMNLP-CoNLL'12*. ACL, 1135–1145.

[14] Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling Relations and Their Mentions without Labeled Text. In *ECML/PKDD'10*.

[15] Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M Marlin. 2013. Relation Extraction with Matrix Factorization and Universal Schemas.. In *HLT-NAACL'13*. ACL, 74–84.

[16] Michael Schmitz, Robert Bart, Stephen Soderland, Oren Etzioni, et al. 2012. Open language learning for information extraction. In *EMNLP-CoNLL'12*. ACL, 523–534.

[17] Jingbo Shang, Jialu Liu, Meng Jiang, Xiang Ren, Clare R Voss, and Jiawei Han. 2017. Automated phrase mining from massive text corpora. *arXiv preprint arXiv:1702.04457* (2017).

[18] Chao Zhang, Guangyu Zhou, Quan Yuan, Honglei Zhuang, Yu Zheng, Lance Kaplan, Shaowen Wang, and Jiawei Han. 2016. Geoburst: Real-time local event detection in geo-tagged tweet streams. In *SIGIR'16*. ACM, 513–522.