Relation Extraction Using Supervision from Topic Knowledge of Relation Labels

Haiyun Jiang[1], **Li Cui**[2], **Zhe Xu**[1], **Deqing Yang**[2], **Jindong Chen**[1], **Chenguang Li**[1], **Jingping Liu**[1], **Jiaqing Liang**[1], **Chao Wang**[1], **Yanghua Xiao**[1,3*] and **Wei Wang**[1,3]

[1]Shanghai Key Laboratory of Data Science, School of Computer Science, Fudan University, China
[2]School of Data Science, Fudan University, Shanghai, China
[3]Shanghai Institute of Intelligent Electronics & Systems, Shanghai, China

Abstract

Explicitly exploring the semantics of a relation is significant for high-accuracy relation extraction, which is, however, not fully studied in previous work. In this paper, we mine the topic knowledge of a relation to explicitly represent the semantics of this relation, and model relation extraction as a matching problem. That is, the matching score between a sentence and a candidate relation is predicted for an entity pair. To this end, we propose a deep matching network to precisely model the semantic similarity between a sentence-relation pair. Besides, the topic knowledge also allows us to derive the importance information of samples as well as two knowledge-guided negative sampling strategies in the training process. We conduct extensive experiments to evaluate the proposed framework and observe improvements in AUC of 11.5% and max F1 of 5.4% over the baselines with state-of-the-art performance.

1 Introduction

Relation extraction (RE) aims to identify the target relation between two entities from a sentence, where the relation set is pre-defined. For example, the relation `founder` will be detected for [*Microsoft, Bill Gates*] given the sentence "Bill Gates co-founded Microsoft with his childhood friend Paul Allen". The popular and reliable solutions for RE are *supervised* and *distantly supervised* paradigms. The traditional supervised methods mainly contain feature-based methods [Alicante and Corazza, 2011] and kernel-based methods [Bunescu and Mooney, 2005] etc. However, the supervised paradigm usually requires expensive annotated data. To overcome this weakness, distant supervision [Mintz *et al.*, 2009] is introduced to automatically build training datasets. Nevertheless, this heuristic approach introduces some noise into the training data, which motivates lots of work to solve the noise problem in distant supervision [Zeng *et al.*, 2015; Lin *et al.*, 2016; Ji *et al.*, 2017].

Existing work under supervised and distantly supervised paradigms mainly studies how to extract informative features from sentences for RE by hand-crafted feature engineering or implicit feature learning using deep learning. In these methods, the relation of a sentence is usually used as a label (i.e., *relation label*) to specify the class of the sentence in the training process, where the label is simply replaced by a meaningless ID. However, these approaches ignore the rich semantics of the relation label that can be used as additional supervision for RE. In fact, exploring the additional supervision from class labels has been concerned in other machine learning areas, e.g., exploiting the class label correlation in computer vision based on statistics [Nguyen and Nguyen, 2019]. But to the best of our knowledge, there are few references to explore *the semantic information* of relation labels in RE task. We argue that obtaining the explicit semantic representation of a relation will provide stronger supervision for high-performance RE.

Intuitively, just a relation label cannot provide enough semantic information, so it is necessary to introduce more external information as background knowledge for a relation. However, the background knowledge is difficult to be obtained since there is no existing knowledge base or other resources that contain a detailed description and explanation of a relation. To overcome this challenge, in this paper we mine the *topic knowledge* of a relation from the training data by conducting topic modeling on the sentence collection labeled with this relation. The basic assumption is that *the sentence collection of a relation contains several latent topics and these topics are semantically related to the relation.* By topic modeling, the top k weighted topic words are extracted to represent the semantics of the relation. In this way, the topic knowledge of a relation is embodied as a weighted bag of words (WBoW). In general, each topic word describes some *aspect* of the relation and the word weight qualifies its importance to the relation. Therefore, a sentence that expresses a target relation is required to match several important aspects of this relation.

We present an example in Figure 1 to illustrate how topic knowledge provides strong supervision for RE. Given a sentence on the left in Figure 1, we hope to infer whether the sentence is expressing the relation `the-CEO-of`. For human beings, we conclude that the first and second sentences explicitly express the relation while the third sentence weakly expresses it. Moreover, the fourth sentence does not express the relation. By introducing the topic knowledge for

*Contact Author. shawyh@fudan.edu.cn

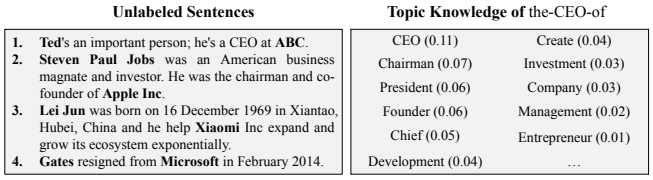

Figure 1: An example of describing the effectiveness of the supervision provided by topic knowledge. Each sentence on the left contains an entity pair and the relation is unknown. The right are the topic words of the relation `the-CEO-of` with an importance weight in the bracket.

`the-CEO-of`, we find that the first and second sentences match most of the important topic words while the final sentence hardly matches any one of these. In this way, the semantic information provided by topic knowledge strongly supports the relation inference.

In this paper, we incorporate the topic knowledge to improve RE from the following three aspects: (a) deep sentence-relation matching, (b) sample reweighting and (c) negative sampling.

Deep sentence-relation matching. We model RE as a *matching* problem. Given an entity pair, the input is a sentence-relation pair and the output is the corresponding matching score, where the relation is represented by its topic words. Instead of shallow word-level matching, we build a deep matching network. Specifically, we take the self-attention mechanism [Vaswani *et al.*, 2017] to extract informative features from a sentence as its final representation. Besides, the *weighted* self-attention mechanism is used to learn the representation of the topic words, where the priori weight distribution in the WBoW is considered. Finally, the interaction between the sentence-relation pair is conducted to capture the global matching features, thus obtaining the final matching score.

Sample reweighting. In this work, we propose a novel reweighting scheme based on topic knowledge to highlight the high-quality samples with a large weight in the training process. The basic idea is to pre-estimate the *semantic distance* between the sentence and the topic knowledge of the relation in a sample using Word Mover's Distance (WMD) [Kusner *et al.*, 2015]. This distance provides evidence for evaluating the sample weight.

Negative sampling. We also propose two knowledge-guided negative sampling strategies based on semantic distance using topic knowledge. In general, the model should pay more attention to the *"difficult" negative samples that have a smaller semantic distance* indicating the sentence has a certain similarity with the topic knowledge of a target relation but it does not express the relation. In this way, the topic knowledge provides strong evidence to identify the most challenging negative samples from the training set, thus fewer negative samples are needed for effective training.

Note that the proposed framework can be easily applied to many classification-based NLP tasks, e.g., long document classification and sentiment analysis, where the topic words of a class label are mined from the training data. In our experiments, we verify the validity of the topic knowledge for RE on the widely used NYT [Riedel *et al.*, 2010] dataset. Furthermore, all the components of the framework will be analyzed separately to prove their effectiveness.

Contributions. Our contributions are three-fold. (1) The idea of exploiting relation semantics as supervision is proposed for RE. (2) We propose a deep sentence-relation matching network to model the semantic similarity for RE. (3) The topic knowledge is also used to derive the importance information of samples, thus helping sample reweighting and negative sampling.

2 Previous Work

The early solutions to supervised RE mainly fall into two categories: feature-based methods [Alicante and Corazza, 2011] and kernel-based methods [Bunescu and Mooney, 2005]. The former extracts different types of features (e.g., lexical, syntactic, as well as semantic ones) and feed them into a classifier. The latter aims to design reasonable kernel functions for RE. The supervised methods usually require expensive labeled data. To overcome it, distant supervision is introduced to automatically generate large amounts of training data [Mintz *et al.*, 2009]. [Hoffmann and Zhang, 2011; Riedel *et al.*, 2010] model RE as a multi-instance learning problem to address the noise problem in distant supervision and propose a probabilistic graphical model to select the best sentence for an entity pair. [Alfonseca *et al.*, 2012] views a syntactic or lexical pattern p as a topic word for a target relation r and proposes a hierarchical topic model, thus predicting $p(r|p)$ with a large score for the informative patterns.

In recent years, the deep learning-based RE has been extensively studied, which improves the performance significantly. [Zeng *et al.*, 2014] adopts an end-to-end convolutional neural network (CNN) for sentence embeddings, and [Zeng *et al.*, 2015] further proposes a piece-wise (three-segment) CNN with multi-instance learning to obtain sentence embeddings. [Lin *et al.*, 2016] proposes the sentence-level attention to highlight the informative sentences, which shows promising results. In [Zheng *et al.*, 2017], entities and relations are jointly extracted in one model with a novel tagging scheme. Besides, [Feng *et al.*, 2018] take reinforcement learning to select the clean sentence for the classifiers.

Some existing work also exploits external knowledge to improve RE. [GuoDong *et al.*, 2005] explores the semantic knowledge in WordNet as additional features for RE. [Chang *et al.*, 2011] first clusters 7000 relations in Freebase into a set of relation topics at multiple scales and then model the relationship between relations. Finally, the relation for a new sentence is identified by mapping the sentence to the relation topic space. [Rocktäschel *et al.*, 2015] takes the first-order logic knowledge for entity pairs to help matrix factorization-based embeddings. APCNN+D [Ji *et al.*, 2017] uses entity descriptions as background knowledge via an attention layer to improve entity representation and help RE. However, there is no work to explore the semantic information of a relation, a new opportunity to further improve the performance.

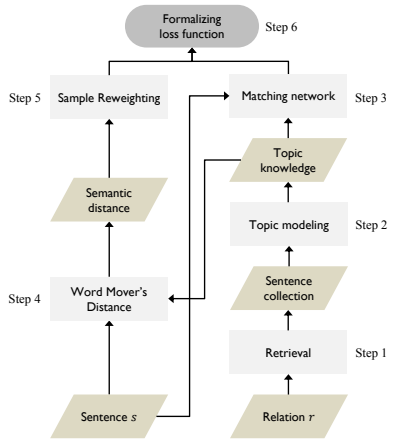

Figure 2: The overview of the proposed topic knowledge-based framework.

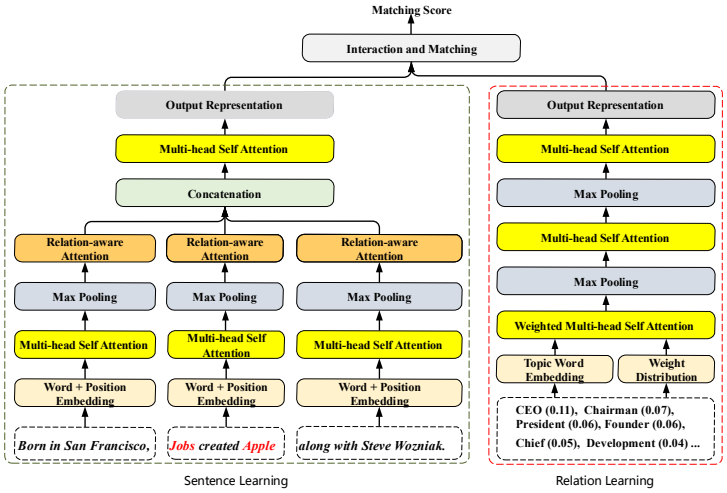

Figure 3: The proposed deep sentence-relation matching network. The left is the sentence learning module and the right is the relation learning module.

3 Framework

3.1 Overview

In this paper, we model RE as a matching problem based on the topic knowledge. Formally, we denote the relation set as $\mathcal{R}$. A training sample is denoted as (x, y) and $x = \langle s, r, t \rangle$, where t is an entity pair, s is a sentence containing t and $r \in \mathcal{R}$ is a candidate relation. $y \in \{0, 1\}$ is the ground-truth label of x. A positive sample (i.e., $y = 1$) indicates that the sentence s expresses the relation r between the two entities in t, otherwise it is a negative sample (i.e., $y = 0$). Our goal is to learn a matching function $p(y|x)$ so that the test samples will be predicted with a correct score.

The topic knowledge of relations helps us to build an accurate and robust matching function. We present the flow chart of our framework in Figure 2. The main steps are as follows. Step 1: Retrieving all the sentences of relation r from the training sentence set. Step 2: Obtaining the topic knowledge for r, that is, extracting the top k weighted topic words from the labeled sentence collection of r by topic modeling. Step 3: Building the deep matching network for a sentence-relation pair (s, r) (Sec 3.2). Step 4: Computing the semantic distance between s and r by Word Mover's Distance (WMD) based on topic knowledge. Step 5: Deriving the importance weight of a sample based on the semantic distance (Sec 3.3). Step 6: Formalizing the loss function to learn the matching network, where the sample weights are incorporated (Sec 3.4). Besides, we also take the topic knowledge to efficiently select fewer negative samples (Sec 3.3). In general, the sample reweighting and knowledge-guided negative sampling help to learn a more efficient and robust matching network.

3.2 Sentence-Relation Matching Network

The proposed matching network is presented in Figure 3, which mainly contains three parts: sentence representation, relation representation and sentence-relation interaction. The modules of sentence learning and relation learning try to extract informative features from a sentence and a relation, respectively. Then the sentence-relation interaction aims to compute the matched score between the extracted features. Each layer in our network is followed by layer normalization operation. In detail, our network consists of the following layers:

Sentence embedding layer. Considering a sentence s consisting of m words $\{w_1, ..., w_m\}$, we first divide the sentence into three blocks by the two entities (as shown in Figure 3, the sentence is divided by *Jobs* and *Apple*), where the length of each block is fixed to l by zero-padding or truncating. This is motivated by the fact that *each block has different importance for relation inference* [Zeng *et al.*, 2015]. Besides, this operation will also significantly reduce the memory requirements and improve the encoding efficiency [Shen *et al.*, 2018]. For example, the complexity of self-attention operation on s is proportional to m^2 [Vaswani *et al.*, 2017]. By dividing s into three blocks and conducting the self-attention on each block in parallel, the complexity is reduced to $3l^2$.

The embedding of a word x in the sentence s is the concatenation $[x_w; x_p] \in \mathbb{R}^{d_1+d_2}$, where $x_w \in \mathbb{R}^{d_1}$ and $x_p \in \mathbb{R}^{d_2}$ are the word embedding and position embedding, respectively. The construction of the position matrix is the same as that in [Zeng *et al.*, 2015], which helps the network to keep track of how close each word is to head or tail entities [Lin *et al.*, 2016]. We denote the embedding of the i-th block as $S_i \in \mathbb{R}^{(d_1+d_2)\times l}$.

Relation embedding layer. This layer aims to generate the embedding of a relation r based on the topic knowledge $\mathcal{A}_r = \{x_1(w_1), ..., x_c(w_c)\}$, where x_i is the i-th topic word with weight w_i. We denote the embedding matrix of the topic word set $\{x_1, ..., x_c\}$ as $C \in \mathbb{R}^{d_1 \times c}$. Different from a sentence, a bag of words does not contain position information. Besides, we also encode the weight distribution $\{w_1, ..., w_c\}$ into a diagonal matrix $W_C \in \mathbb{R}^{c \times c}$ with $(W_C)_{ii} = w_i$, which will guide the representation learning of r.

Multi-head self-attention layer. The self-attention mechanism was successfully applied to model the meaning of sentences in machine translation [Vaswani *et al.*, 2017], reading

comprehension [Shen *et al.*, 2018], etc. Self-attention captures the long-term dependence in a word sequence by learning the interactions between any two words in this sequence. The multi-head setting allows the model to compute the attention weights in different subspaces at different positions [Vaswani *et al.*, 2017]. The attention layer in this paper is the same as that in [Vaswani *et al.*, 2017] with head number $h = 4$.

Weighted multi-head self-attention layer. The standard self-attention assumes each token in a sequence is equally important as inputs. However, in our setting each topic word has a different importance weight for the relation learning, so *the attention weight of the word i to j should not only consider the similarity between them but also incorporate the priori weight of j*. In this way, the final representation of r will pay more attention to the important topic words while still capturing the word dependency. We retain the multi-head setting and modify the self-attention as follows.

$$Attention'(Q, K, V, W_C) = [softmax(\frac{QK^T W_C}{\sqrt{d_1}})V] \quad (1)$$

where $(QK^T W_C)_{ij} = (q_i k_j^T)w_j$ is the weighted attention value of q_i (the i-th row vector in Q) to k_j (the j-th row vector in K), which integrates both the automatically learned attention weight $q_i^T k_j$ and the priori weight w_j.

Max pooling layer. In RE task, most of the sentences have long and complex structures, but only a small span is helpful for relation inference [Zhu *et al.*, 2018]. Max pooling is used to refine the hidden representation and extract the most important features. Given an input $X \in \mathbb{R}^{d\times n}$, max pooling selects the largest $k(k < n)$ values in each row and the output is $Y \in \mathbb{R}^{d\times k}$.

Relation-aware attention layer. We hope to extract the relation-aware features from the hidden representation of a sentence s. Given the hidden representation of a block $X \in \mathbb{R}^{d\times k}$, we take r as a query and conduct the multi-head attention by setting $Q = W^l C W^r \in \mathbb{R}^{d\times k}$ and $K, V = X$, where Q is the transformation of the topic word matrix C and $W^l \in \mathbb{R}^{d\times d_1}$, $W^r \in \mathbb{R}^{c\times k}$ are parameters. We denote the output of this layer as $Y \in \mathbb{R}^{d\times k}$.

Concatenation layer. This layer merges the multiple inputs by $Y = [X_1, ..., X_n]$ with $X_i \in \mathbb{R}^{d\times k}$ and $Y \in \mathbb{R}^{d\times nk}$.

Interaction and matching layer. This layer conducts the interaction between a sentence-relation pair s, r given an entity pair t. We denote the representations of s and r as $O_s \in \mathbb{R}^{d\times b}$ and $O_r \in \mathbb{R}^{d_1\times b_1}$, respectively. The interaction result is presented as a matching score, i.e.,

$$sim(s, r|t) = w^T \tanh(sum(W_1 O_s) + sum(W_2 O_r) + b_1) \quad (2)$$

$$p(y = 1|\langle s, r, t\rangle) = \frac{1}{1 + e^{-sim(s,r|t)}} \quad (3)$$

where $sum(\cdot)$ sums the elements of a matrix into a single column vector, e.g., $sum(\begin{bmatrix}1 & 2\\ 3 & 4\end{bmatrix}) = \begin{bmatrix}3\\ 7\end{bmatrix}$. $W_1 \in \mathbb{R}^{d'\times d}$, $W_1 \in \mathbb{R}^{d'\times d_1}$ and $w, b_1 \in \mathbb{R}^{d'}$ are parameters.

3.3 Sample Reweighting and Negative Sampling

To learn a high-accuracy matching network $p(y|x)$, it is necessary to obtain a dataset with high quality, which is, however, cannot be always satisfied in RE task (especially under the setting of distant supervision). To overcome this problem, we evaluate the *quality of samples* in advance and then reweigh samples proportional to the sample quality in the training process. Moreover, the sample quality also guides the effective negative sampling, a tough problem in most learning-based tasks.

Semantic Distance

Based on the topic knowledge, the evaluation of sample quality can be easily achieved by semantic distance. We present a sentence s as a weighted bag of words (WBoW) $\mathcal{B}_s$, where the weight d_i is the normalized occurrence of word i. To be precise, if word i appears c_i times in the sentence, we denote $d_i = \frac{c_i}{\sum_{j=1}^{m} c_j}$. Given a sentence-relation pair (s, r), the semantic distance between s and r is computed based on $\mathcal{B}_s$ and $\mathcal{A}_r$ (the topic knowledge of r) using Word Mover's Distance [Kusner *et al.*, 2015] and we denote it as $dis(s, r) \geq 0$.

Sample Reweighting

$dis(s, r)$ denotes the semantic relevance between s and r. In general, a positive sample with a small $dis(s, r)$ indicates that s explicitly expresses r and should be given more attention in the training process. Besides, a negative sample with a small $dis(s, r)$ indicates that the sentence s is semantically related to r but it does not express r. These negative samples should also be paid more attention because the similar ones in the test set are easy to deceive the matching network and be wrongly predicted as positive. In contrast, the negative samples with a large $dis(s, r)$ can be easily predicted with a correct score. Therefore, we allocate a weight that is proportional to $-dis(s, r)$ for both the positive and negative samples. Specifically, given a training sample of (s, r), we compute the sample weight as $w = e^{-dis(s,r)}$.

Negative Sampling

In RE task, there are too many negative samples to be considered during the training process, and the direct solution is the random sampling [Riedel *et al.*, 2010]. However, this solution is blind and tends to ignore many informative and valuable samples. Instead, we provide two priori knowledge-guided negative sampling strategies.

We have stated that the negative samples with a small $dis(s, r)$ should be paid more attention because they are difficult to be predicted, in other words, these samples are more informative and valuable as training data. The first sampling strategy is to rank all the negative samples according to $dis(s, r)$ and then select the top k ones with the smallest $dis(s, r)$ as the sampling result. The second strategy is to sample k ones from all the negative samples with the probability proportional to the sample weight w.

3.4 Loss Function

We denote the training samples as $\mathcal{D} = \{(x_i, y_i), i = 1, 2, ...|\mathcal{D}|\}$, where $x_i = \langle s_i, r_i, t_i\rangle$ and $y_i \in \{1, 0\}$ is the ground-truth label of x_i. The loss function is:

$$\mathcal{L}(\theta) = \sum\nolimits_{i=1}^{|\mathcal{D}|} \hat{w}_i \mathcal{L}_B[p(y_i|x_i; \theta)] \quad (4)$$

where the weighted cross entropy is used and $\mathcal{L}_B[p]$ is the binary entropy for p. $\hat{w}_i$ is the normalized weights, that is, $\hat{w}_i =$

$\frac{|\mathcal{D}_p|w_i}{\sum_{x_l \in \mathcal{D}_p} w_l}$ for the positive samples and $\hat{w}_i = \frac{|\mathcal{D}_n|w_i}{\sum_{x_l \in \mathcal{D}_n} w_l}$ for the negative ones, where $\mathcal{D}_p$ and $\mathcal{D}_n$ denote the positive and negative training sample set, respectively.

4 Experiments

We evaluate the proposed framework from four aspects: overall performance (Sec 4.1), performance of semantic distance (Sec 4.2), sample reweighting and negative sampling (Sec 4.3).

Dataset Description

NYT [Riedel *et al.*, 2010] is a distantly supervised RE dataset and is generated by aligning Freebase relations with the New York Times corpus. There are 522,611 labeled sentences, 281,270 entity pairs, and 18,252 relational facts (i.e., $\langle$*head entity, relation, tail entity*$\rangle$) in the training set; and 172,448 sentences, 96,678 entity pairs and 1,950 relational facts in the test set. The dataset contains 53 unique relations from Freebase including a special relation "NA" that denotes no relation between the two entities in a sentence.

Different from the multi-class classification based methods (where "NA" is one of the classes), the samples for training our model should not contain "NA". Because it is not a *specific* semantic relation and has no topic knowledge. Therefore, we have to reconstruct the training samples based on the original data. Specifically, a positive sample is a sentence-relation pair where the sentence is labeled with this relation (except for "NA") in the training set of NYT. To derive the negative sample set, we label each sentence of "NA" with all the 53 unique relations respectively and each sentence will generate 53 negative samples. In this way, we obtain 109,187 positive and 22,070,427 negative samples. The negative sampling strategies will be used to select a subset from this negative set. Note that *the evaluation metrics are the same* for all models in comparison (see Sec 4.1).

Training and Testing Details

In all experiments, we use the word and position embeddings trained by [Lin *et al.*, 2016] with word dimension $d_1 = 50$ and position dimension $d_2 = 5$. During training, the dropout between layers is used with a dropout rate of 0.1. The loss function is minimized by mini-batch gradient descent, where the batch size is 50, and the learning rate is 0.005.

During testing, we use the learned $p(y|x)$ to predict the matching scores for all relations given the sentence s of an entity pair t. In this way, we extract the relations $\{r'\}$ with $p(y = 1|\langle s, r', t\rangle) > \delta$ (a predefined threshold) for t. Furthermore, when there is more than one sentence containing the same entity pair t, the relation r' will be extracted if there is at least one sentence s' that makes $p(y = 1|\langle s', r', t\rangle) > \delta$.

4.1 Overall Performance Evaluation

We evaluate the overall performance of the proposed framework and compare it with the existing work.

Parameter Settings. We conduct LDA based topic modeling on the sentence collection of a relation. More specifically, the topic distribution θ and the word distribution β are subjected to Dirichlet distribution. The top-c topic words among all the topics were selected by computing $p(t)p(w|t)$, where $p(t)$ is the topic distribution and $p(w|t)$ is the word distribution given topic t. Note that the setting with a large number of topics will generate some unrelated topics. In general, our experiments find that almost all the generated topics are semantically related to the target relation when topic number is 5 with other default parameters.

The ratio of positive and negative samples is set to 1:4 and we take the second negative sampling strategy to obtain 636,748 ones from the negative sample set. We set $k = 4$ in the max pooling layer for the sentence learning and $k = 20$ (10) in the first (second) pooling layer for the relation learning. The other untrainable parameters in our framework are tuned by grid search using three-fold validation on the training set, where the topic word size $c \in \{20, 30, 50, 70\}$, the block length $l \in \{5, 6, 7, 8, 9, 10\}$, $d' \in \{60, 65, 70, 75, 80\}$. The optimal settings are $c = 50$, $l = 7$ and $d' = 70$.

Baselines. We consider 8 strong baselines for comparison. (1) The *PCNN* model proposed by [Zeng *et al.*, 2015] is a variation of CNN and it adopts piecewise max pooling in sentence learning. (2) *Logic-MF* by [Rocktäschel *et al.*, 2015] injects the logical knowledge into the matrix factorization-based embeddings for entity pairs and relations. (3) *PCNN+ATT* is a sentence-level attention-based model proposed by [Lin *et al.*, 2016], which uses PCNN in [Zeng *et al.*, 2015] to learn the sentence representations. (4) *MIML-CNN* [Jiang *et al.*, 2016] is also a CNN-based multi-instance multi-label framework, where the cross-sentence max-pooling is used to extract the max features over a bag. (5) *APCNN+D* is an attention-based model proposed by [Ji *et al.*, 2017], where the entity descriptions are introduced to enhance the inputs. (6) and (7): The idea of adversarial training is introduced into RE task by [Wu *et al.*, 2017], which includes two models: *PCNN+Adv* and *RNN+Adv*. (8) CNN+ATT+RL [Feng *et al.*, 2018] first selects the true positive instances by reinforcement learning and then train the PCNN+ATT model using the selected clean samples. Besides, we abbreviate our **T**opic **K**nowledge-based **M**atching **F**ramework as *TK-MF*.

Evaluation Metrics. Both the *held-out evaluation* [Zeng *et al.*, 2015; Lin *et al.*, 2016; Feng *et al.*, 2018; Ji *et al.*, 2017] and *manual evaluation* [Mintz *et al.*, 2009; Zeng *et al.*, 2015; Ji *et al.*, 2017] are conducted. The former compares the predicted relations of the entity pairs in the test set with the ground-truth ones, where the precision, recall, AUC and F1-score are used as the metrics. However, the ground-truth relations of an entity pair may be lost or wrong in the test set, so the held-out evaluation may cause unfair comparison. The latter aims to *manually* evaluate model performance by five volunteers, where the accuracy for the top k predicted facts is computed. More details of manual evaluation can be found in [Zeng *et al.*, 2015; Ji *et al.*, 2017].

Results and Analysis. The held-out and manual evaluation results are presented in Tables 1 and 2, respectively. We conclude that our framework (i.e., TK-MF) with $c = 50$ improves the AUC of 11.5% compared with APCNN+D and $\max$ F1 of 5.4% compared with PCNN+ATT. The results indicate that the introduction of topic knowledge significantly improves the performance of RE compared with the methods that implicitly learn the knowledge of a relation. Besides, we also

Recall	0.05	0.01	0.15	0.20	0.25	0.30	0.35	AUC	max F1
PCNN	0.82	0.79	0.71	0.68	0.61	0.55	0.48	0.348	0.405
Logic-MF	0.79	0.75	0.68	0.66	0.63	0.53	0.49	0.360	0.408
PCNN+ATT	0.83	0.78	0.71	0.66	0.62	0.59	0.53	0.387	0.422
MIML-CNN	0.71	0.65	0.61	0.57	0.53	0.48	0.44	0.316	0.389
APCNN+D	0.78	0.76	0.72	0.65	0.62	0.58	0.51	0.390	0.415
PCNN+Adv	0.86	0.71	0.68	0.61	0.57	0.52	0.46	0.315	0.398
RNN+Adv	0.83	0.78	0.62	0.58	0.53	0.50	0.48	0.305	0.405
CNN+ATT+RL	0.85	0.73	0.68	0.67	0.59	0.57	0.52	0.369	0.418
TK-MF($c = 50$)	**0.94**	**0.84**	0.75	**0.71**	**0.69**	**0.65**	**0.63**	**0.435**	**0.445**
TK-MF($c = 20$)	0.90	**0.84**	0.74	0.63	0.57	0.52	0.48	0.423	0.433
TK-MF($c = 30$)	0.93	0.83	**0.76**	0.64	0.57	0.51	0.45	0.421	0.426
TK-MF($c = 70$)	0.76	0.56	0.49	0.41	0.36	0.33	0.31	0.295	0.334

Table 1: The sampled recall-precision pairs, AUCs and max F1s for different models. "c" is the size of the topic words for each relation in our model.

Model	Top 100	Top 200	Top 500	Average
PCNN	0.86	0.80	0.69	0.783
Logic-MF	0.88	0.79	0.66	0.776
PCNN+ATT	0.87	0.82	0.73	0.806
MIML-CNN	0.82	0.76	0.71	0.763
APCNN+D	0.87	0.83	0.74	0.816
PCNN+Adv	0.87	0.82	0.71	0.804
RNN+Adv	0.87	0.83	0.72	0.813
CNN+ATT+RL	0.85	0.82	0.74	0.802
TK-MF($c = 50$)	**0.93**	**0.88**	**0.80**	**0.870**
TK-MF($c = 20$)	0.91	0.86	0.76	0.843
TK-MF($c = 30$)	0.92	0.87	0.77	0.853
TK-MF($c = 70$)	0.87	0.81	0.72	0.800

Table 2: Accuracies for the top 100, 200, and 500 predicted facts upon manual evaluation.

c	Top-1	Top-3	Top-5	Top-10	Top-15	Top-20	MRR
20	0.223	0.416	0.537	0.699	0.785	0.843	0.371
30	0.213	0.369	0.479	0.670	0.773	0.845	0.351
50	0.232	0.398	0.498	0.690	0.794	0.856	0.373
70	0.128	0.227	0.319	0.462	0.582	0.700	0.234

Table 3: The recalls for different k as well as the MRR values, where c is the topic word size.

present the results with different sizes of topic words in the tables, which shows a proper size is required for high performance. This is because the topic knowledge with a small size (e.g., $c = 20$) is unable to capture all the topic words of a relation while a large size (e.g., $c = 70$) will introduce many unrelated words.

4.2 Evaluation of Semantic Distance

The performance of semantic distance is very important in sample reweighting and negative sampling. We evaluate the topic knowledge-based distance as follows. Given the sentence s from a positive sample in NYT, we consider all the relations $\mathcal{R}$ and compute the distance between s and $r' \in \mathcal{R}$. Then we rank the distances in *ascending* order and check whether the true relation r is in the top k relation set. We evaluate the performance by $recall = \frac{\sum I_i}{N}$, where $N = 10000$ is the number of objects to be evaluated and $I_i \in \{0, 1\}$ indicating whether the true relation is in the top k relation set for the i-th object. Besides, the mean reciprocal rank (MRR) is also computed by $MRR = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{rank_i}$, where $rank_i$ is the ranking of the true relation in the i-th sample based on the distance in ascending order.

The results are presented in Table 3. We conclude that the true relation is ranked in front of most of the false ones for a given sentence in general. For example, the MRR is 0.373 for $c = 50$, which means the average ranking of a positive relation is about 2.68 (i.e., 1/0.373) among 53 relations.

4.3 Evaluation of Sample Reweighting and Negative Sampling

We evaluate the performance improvements introduced by sample reweighting and knowledge-guided negative sampling on the NYT dataset. We present the results in Table 4, where the untrainable parameters are the same as those in section 4.1. We conclude that the results with sample reweighting is better than the ones without reweighting under the same settings. Besides, both the two proposed sampling strategies outperform random sampling in performance. Moreover, the

Reweighting	Sampling	$\lvert\mathcal{D}_p\rvert$:$\lvert\mathcal{D}_n\rvert$	AUC	max F1
Yes (No)	Random	1:1	0.361 (0.351)	0.371 (0.362)
Yes (No)	Random	1:2	0.374 (0.363)	0.372 (0.373)
Yes (No)	Random	1:4	0.383 (0.362)	0.376 (0.374)
Yes (No)	First	1:1	0.404 (0.361)	0.415 (0.374)
Yes (No)	First	1:2	0.416 (0.371)	0.402 (0.380)
Yes (No)	First	1:4	0.421 (0.393)	0.422 (0.401)
Yes (No)	Second	1:1	0.427 (0.392)	0.424 (0.397)
Yes (No)	Second	1:2	0.431 (0.388)	0.430 (0.401)
Yes (No)	Second	1:4	0.435 (0.402)	0.445 (0.412)

Table 4: The held-out results on the NYT dataset. "First" and "Second" denote the two different sampling strategies, respectively. $|\mathcal{D}_p|$ and $|\mathcal{D}_n|$ are the sizes of positive and negative samples, respectively.

second strategy outperforms the first one in general, which is caused by the fact that the second strategy is the *trade-off* between the first strategy and the randomly sampling. That is, the second strategy not only considers the sample importance but also retains the sampling diversity. Finally, we note that the results by two strategies with $|\mathcal{D}_p|$:$|\mathcal{D}_n| =$ 1:1 are also better than those by random sampling with $|\mathcal{D}_p|$:$|\mathcal{D}_n| =$ 1:4 or 1:2, which indicates that our strategies can effectively select *fewer and informative* samples compared with random sampling.

5 Conclusion

In this paper, we explore the semantics of a relation with topic knowledge, which provides strong supervision for relation extraction. A novel deep matching network is further proposed to detect the relations of entity pairs based on topic knowledge. Meanwhile, we incorporate the sample reweighting into training and introduce two knowledge-guided negative sampling strategies. Our framework is suitable for many classification-based NLP tasks, which is also the major research direction in our future work.

Acknowledgments

This work was supported by National Key R&D Program of China (No.2015CB35881,2017YFC1201203), Shanghai Municipal Science and Technology Major Project (Grant No 16JC1420400) and NSFC Project (No.61732004 and U1509213).

References

[Alfonseca *et al.*, 2012] Enrique Alfonseca, Katja Filippova, Jean Yves Delort, and Guillermo Garrido. Pattern learning for relation extraction with a hierarchical topic model. In

Meeting of the Association for Computational Linguistics: Short Papers, 2012.

[Alicante and Corazza, 2011] Anita Alicante and Anna Corazza. Barrier features for classification of semantic relations. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pages 509–514, 2011.

[Bunescu and Mooney, 2005] Razvan C Bunescu and Raymond J Mooney. A shortest path dependency kernel for relation extraction. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 724–731. Association for Computational Linguistics, 2005.

[Chang *et al.*, 2011] Wang Chang, James Fan, Aditya Kalyanpur, and David Gondek. Relation extraction with relation topics. In *Conference on Empirical Methods in Natural Language Processing*, 2011.

[Feng *et al.*, 2018] Jun Feng, Minlie Huang, Li Zhao, Yang Yang, and Xiaoyan Zhu. Reinforcement learning for relation classification from noisy data. In *Proceedings of AAAI*, 2018.

[GuoDong *et al.*, 2005] Zhou GuoDong, Su Jian, Zhang Jie, and Zhang Min. 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, 2005.

[Hoffmann and Zhang, 2011] Raphael Hoffmann and Congle Zhang. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 541–550. Association for Computational Linguistics, 2011.

[Ji *et al.*, 2017] Guoliang Ji, Kang Liu, Shizhu He, Jun Zhao, et al. Distant supervision for relation extraction with sentence-level attention and entity descriptions. In *AAAI*, pages 3060–3066, 2017.

[Jiang *et al.*, 2016] Xiaotian Jiang, Quan Wang, Peng Li, and Bin Wang. Relation extraction with multi-instance multi-label convolutional neural networks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics*, pages 1471–1480, 2016.

[Kusner *et al.*, 2015] Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *International Conference on Machine Learning*, pages 957–966, 2015.

[Lin *et al.*, 2016] Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 2124–2133, 2016.

[Mintz *et al.*, 2009] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL*, pages 1003–1011. Association for Computational Linguistics, 2009.

[Nguyen and Nguyen, 2019] Tien Thanh Nguyen and Thi Thu Thuy Nguyen. Multi-label classification via label correlation and first order feature dependance in a data stream. *Pattern Recognition*, 90:35–51, 2019.

[Riedel *et al.*, 2010] Sebastian Riedel, Limin Yao, and Andrew McCallum. Modeling relations and their mentions without labeled text. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer, 2010.

[Rocktäschel *et al.*, 2015] Tim Rocktäschel, Sameer Singh, and Sebastian Riedel. Injecting logical background knowledge into embeddings for relation extraction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1119–1129, 2015.

[Shen *et al.*, 2018] Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, and Chengqi Zhang. Bi-directional block self-attention for fast and memory-efficient sequence modeling. *arXiv preprint arXiv:1804.00857*, 2018.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[Wu *et al.*, 2017] Yi Wu, David Bamman, and Stuart Russell. Adversarial training for relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1778–1783, 2017.

[Zeng *et al.*, 2014] Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics*, pages 2335–2344, 2014.

[Zeng *et al.*, 2015] Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 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, 2015.

[Zheng *et al.*, 2017] Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. Joint extraction of entities and relations based on a novel tagging scheme. *arXiv preprint arXiv:1706.05075*, 2017.

[Zhu *et al.*, 2018] Wenhao Zhu, Tengjun Yao, Jianyue Ni, Baogang Wei, and Zhiguo Lu. Dependency-based siamese long short-term memory network for learning sentence representations. *PloS one*, 13(3):e0193919, 2018.