Improving Neural Relation Extraction with Implicit Mutual Relations

Jun Kuang*, Yixin Cao†, Jianbing Zheng*, Xiangnan He‡, Ming Gao*§, Aoying Zhou*

*School of Data Science and Engineering, East China Normal University, Shanghai, China
†School of Computing, National University of Singapore, Singapore
‡School of Information Science and Technology, University of Science and Technology of China, Hefei, China
§KLATASDS-MOE, School of Statistics, East China Normal University, Shanghai, China

***Abstract*—Relation extraction (RE) aims at extracting the relation between two entities from the text corpora. It is a crucial task for Knowledge Graph (KG) construction. Most existing methods predict the relation between an entity pair by learning the relation from the training sentences, which contain the targeted entity pair. In contrast to existing distant supervision approaches that suffer from insufficient training corpora to extract relations, our proposal of mining implicit mutual relation from the massive unlabeled corpora transfers the semantic information of entity pairs into the RE model, which is more expressive and semantically plausible. After constructing an entity proximity graph based on the implicit mutual relations, we preserve the semantic relations of entity pairs via embedding each vertex of the graph into a low-dimensional space. As a result, we can easily and flexibly integrate the implicit mutual relations and other entity information, such as entity types, into the existing RE methods.**

Our experimental results on a New York Times and another Google Distant Supervision datasets suggest that our proposed neural RE framework provides a promising improvement for the RE task, and significantly outperforms the state-of-the-art methods. Moreover, the component for mining implicit mutual relations is so flexible that can help to improve the performance of both CNN-based and RNN-based RE models significant.

***Index Terms*—Relation extraction, implicit mutual relations, unlabeled data, entity information**

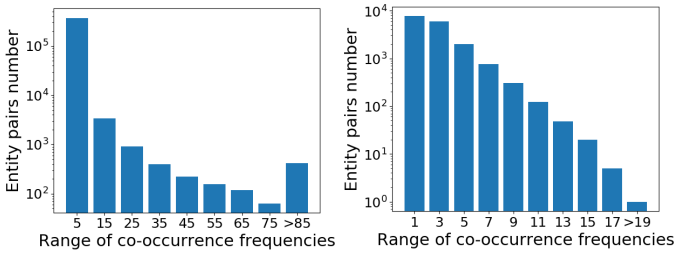
I. INTRODUCTION

Recently, we have witnessed an ocean of Knowledge Graphs (KGs), such as DBpedia [1], FreeBase [2], and YAGO [3], which has been successfully applied in a host of tasks, including question answering [4], search engine [5] and chat robot [6]. These KGs are far from complete. Thus, it attracts much attention to extract factual triplet from plain text for KG completion, e.g., (**Obama**, *born*, **Hawaii**), which involves two sub-tasks of entity linking [7] and relation extraction (RE) [8].

As a paramount step, RE is typically regarded as a classification problem [9]. Given two entities (e.g., **Obama** and **Hawaii**), RE aims at classifying them into pre-defined relation types (e.g., *born*) based on the sentences involving the entity pair. It is nontrivial since the same relation may have various textual expressions, and meanwhile, different relations can also be described using the same words.

Existing RE approaches have achieved a great success based on deep neural network (NN) [10], [11]. They encode the texts via CNN [10] or RNN [11] without feature engineering, then feed the hidden states into a softmax layer for classification. However, there are two issues arise from NN-based RE models:

Insufficient Training Corpora For satisfactory performance, these NN-based models require a large amount of training data, which is usually expensive to obtain. Alternatively, distant supervision is proposed to automatically extract sentences for training [12]. It is under the assumption that if two entities $(head, tail)$ participate in a relation r, any sentence that contains $head$ and $tail$ might express that relation. However, there are still many infrequent entity pairs lacking sufficient training data due to the long-tailed distribution of frequencies of entity pairs. As illustrated in Figure 1, we count the number of entity pairs in log-scale with different range of co-occurrence frequencies in the dataset. The x-axis denotes the range of co-occurrence frequencies in the corresponding dataset. The y-axis denotes the number of entity pairs which co-occurrence frequencies are in the corresponding range. We can find that more than 90% of the entity pairs in the GDS dataset have co-occurrence frequencies less than 10, and this situation becomes more severe in NYT dataset.



(a) NYT (b) GDS

Fig. 1. The number of entity pairs with different training data via distant supervision. The y-axis uses the log-scale.

Noisy Data Although a large number of labeled data can be employed to train the RE algorithms with the help of distant supervision, the assumption sometimes is too strong and may introduce much noise. For example, the sentence "Barack Obama visits Hawaii" cannot express the relation *born in*, but distant supervision would take it as ground truth.

TABLE I
AN EXAMPLE OF THE IMPLICIT MUTUAL RELATIONS BETWEEN ENTITY PAIRS.

ID	Entity pair	Sentences	Sentence example	Relation
1	(Stanford University, California)	2	...and the **California**, ...learned from **Stanford University**...	hard to extract
2	(University of Washington, Seattle)	17	...research at the **University of Washington** in **Seattle**...	*locatedIn*
3	(University of Southern California, Los Angeles)	13	...at the **University of Southern California** in **Los Angeles**...	*locatedIn*
4	(Columbia University, New York City)	24	...in **New York City**, ...graduated from **Columbia University**...	*locatedIn*

Existing methods usually alleviate the negative impact of noise by utilizing attention mechanism [13]. They select high-quality sentences by assigning them to higher weights and reduce the impact of noisy sentences through setting lower weights to them. However, we argue that the abandon of sentences may exacerbate the inadequate issue of training data.

By intuition, if different entity pairs are similar in semantic, they are more likely to have the same relation. For example, we illustrate four entity pairs in Table I, where they are similar in semantic, and all have the *locatedIn* relation. For target entity pair ep_1 =(**Stanford University**, **California**), if there are only two sentences in the distant supervision training dataset, its relation is not easily predicted due to the insufficient training instances and noisy data (e.g., the listed sentence in ID1 of Table I cannot express the *locatedIn* relation of ep_1). As illustrated in Figure 1, infrequent entity pairs are very common cases in the distant supervision training datasets. Fortunately, all entity pairs, such as ep_2, ep_3 and ep_4, are helpful to predict the relation of the target entity pair ep_1. Not limited to this, for target entity pair ep_2 =(**University of Washington**, **Seattle**), the semantic information of entity pair ep_3 = (**University of Southern California, Los Angeles**) is similar to the target entity pair ep_2, and therefore helpful to predict the relation of entity pair ep_2, vice versa. In a word, all entity pairs with similar semantic are helpful to extract relation to each other.

However, most of the approaches cannot capture the similar semantic from the training dataset since only sentences, which contain the target entity pair, are employed to train the RE model. Many existing works, such as Word2vec [14], GloVe [15], and BERT [16], etc., can extract the semantic information of words from the unlabeled corpora, rather than entities. In contrast to extracting semantic information of words, we aim at mining the semantic information of entities to furthermore improve the performance of RE model.

To capture the mutual semantic relation between entities, we construct an entity proximity graph based on the unlabeled corpora, and further employ an embedding-based approach to learn a low-dimensional representation of each entity. As a result, the relations between entities are implicitly represented in the low-dimensional and semantic space, where entities with the similar semantic are closed in the space. Thus, we call the relations as **implicit mutual relations** of entities. In particular, the component for mining implicit mutual relations can be seamlessly and flexibly integrated into the other RE models, such as CNN-based and RNN-based approaches.

In addition, the relation *place of birth* must be a relation between two entities whose types are *Location* and *Person*. Entity types are therefore helpful to predict the relation of a target entity pair. Thus, except for implicit mutual relations, we also integrate the distant supervision training data and the other entity information, such as entity types, to further improve the RE model. The main contributions of this paper are as follows:

- We propose to utilize implicit mutual relations between entity pairs to improve the RE task, and we mine such mutual relations from the easily available unlabeled data.
- We design a unified and flexible deep neural network framework, which ensembles the training corpora, entity types and implicit mutual relations, is proposed to extract relation from the plain text.
- We evaluate the proposed algorithm against baselines on two datasets. Experimental results illustrate the promising performance, and indicate that the implicit mutual relations are rewarding to improve the performance of both CNN-based and RNN-based RE models.

The rest of the paper is organized as follows. Section II covers the related works. In Section III, we formulate the problem formally, and provide our solution for RE. We report the promising experiment results on real-world datasets in Section IV. Finally, we conclude the paper in Section V.

II. RELATED WORK

For extracting relations from the training corpora, supervised learning methods are the most effective [17].

Especially, the neural network methods for relation extraction have also made a great progress in recent years. Socher et al. [18] first parse the sentences and then use a recursive network to encode the sentences. Zeng et al. [10] propose a CNN-based model which can capture the lexical and sentence level features. Zeng et al. [19] improve the CNN-based model by using the piecewise max pooling in the pooling layer of CNN.

Lots of works are focusing on improving the performances of the neural network methods, these works mainly start from the following aspects:

- The neural encoder is adopted to extract various features from training corpora, such as syntax, semantics, etc [20] [21]. The encoders which better capture and express the information can lead to better performance on relation extraction. Therefore, many works focus on improving the neural encoder to get more prominent relation extraction models.

- The neural network methods perform well for relation extraction. However, These methods require labor overhead for data annotation. As a result, the problem of lacking labeled data is more serious for large scale datasets. To address the issue, the distant supervision learning is proposed [12]. The distant supervision learning is under the assumption that if an entity pair $(head, tail)$ has a relation r, any sentences that contain $head$ and $tail$ might express this relation. So labeled data can be obtained by aligning training corpora to KGs. However, the distant supervision will inevitably introduce the noise into the training data. Therefore, many works attempt to address how to alleviate the performance loss caused by noisy data [13] [22] [23].
- Some works extract the relations of targeted entity pair only using the text which contains the entities in the target pair, while the others try to improve the relation extraction via mining the extra useful information, such as relation alias information [24], relation path [25], and entity description [26], etc. This extra information can be mined from various sources, including labeled and unlabeled data. The researchers integrate the extra information into relation extraction model as a supplementary, to enrich the information which relation extraction needs.

A. Neural Encoder Improvement

Some works design more sophisticated neural network encoders to improve the performance of relational extraction. Santos et al. [27] use a convolutional neural network that performs relation extraction by ranking(CR-CNN). Nguyen et al. [28] utilize multiple window sizes for CNN filters to obtain more features. Miwa et al. [21] stack bidirectional tree-structured LSTM-RNNs on bidirectional sequential LSTM-RNNs to encode both word sequence and dependency tree substructure information. Moreover, Christopoulou et al. [20] encode multiple entity pairs in a sentence simultaneously to make use of the interaction among them. They place all the entities in a sentence as nodes in a full-connected entity graph, and encode them with a walk-based model on the entity graph.

B. Noise Mitigation

To mitigate the noise in distant supervision learning, some works [22] [23] utilize the multi-instance learning which allows different sentences to have at most one shared label. The multi-instance learning combines all relevant instances to determine the relation of the targeted entity pair, thereby alleviating the impact of wrong labeled instances. Surdeanu et al. [29] get rid of the restrict that different sentences can only share one label by utilizing a graphical model which can jointly model the multiple instances and multiple relations.

With the development of the neural network, a technique called attention mechanism is proposed [30]. The attention mechanism can let neural network models focus on the important training sentences. In the field of relation extraction, attention mechanism is widely used to mitigate the effects of noisy data [31]. Existing attention approaches can be categorized into two groups: sentence-level attention and word-level attention. Sentence-level attention [13] aims at selecting the sentences w.r.t. the relational strength between the target entity pair. Similarly, word-level attention [11] focuses on high-quality words to measure the target relation. Furthermore, some works [32] [33] [34] adopt the hierarchical attention which combines these two attention mechanisms, and further improves the performance of relation extraction.

Alternatively, reinforcement learning can also alleviate the effects of noisy data [35] [36]. The reinforcement learning methods mainly consist of two modules: a module is called instance selector to select the high-quality instances, and the other module is called relation classifier to make the prediction and provide rewards to the instance selector. The noisy data will be eliminated by the instance selector, that leads to a performance improvement.

Adversarial training is also a viable solution to address the noise problem. Wu et al. [37] introduce adversarial training [38] into the relation extraction task. They generate adversarial samples by first adding noise in the form of small perturbations to the original data, then encouraging the neural network to correctly classify both unmodified examples and perturbed ones to regularizing the relation extraction model. The regularized relation extraction model is more robusted and has higher generalization performance, so it can fight noise data very well. Furthermore, Qin et al. [39] utilize the Generative Adaversarial Networks(GANs) [40] to filter distant supervision training dataset and redistribute the false positive instances into the negative set.

C. Extra Information supplementary

The other direction to improve the performance of relation extraction model is to integrate more useful information into the existing approaches. This extra information is a good supplementary because this information cannot be extracted from the training corpora directly.

Some works attempt to introduce extra relation information. [24] et al. utilize the relation alias information (e.g. *founded* and *co-founded* are aliases for the relation *founderOfCompany*) to enhance the relation extraction. Zeng et al. [25] construct the relation path between two entities that are not in the same sentence. Ji et al [26] utilize the entity description information to supplement background knowledge. Liu et al. [41] improve the relation extraction with entity type information. In addition, the semantic information [42] and part-of-speech tag [43] are also good supplementary.

Although the additional information can improve the performance of relation extraction models, some of the information relies on high-quality sources which are expensive to collect. In our solution, we mine the implicit mutual relations between entity pairs from the available unlabeled data. In addition, the entity type information we used is also easily obtained via aligning the training corpora to KGs, which contain the entity type information.

III. METHODOLOGY

Given a target entity pair $(head, tail)$, and a set of training sentences $S = \{s_1, s_2, \cdots, s_n\}$, where each sentence s_i contains the entities $head$ and $tail$. Our model aims at classifying the relation r between entities $head$ and $tail$ by utilizing the sentences, the implicit mutual relations and the entity type information. As illustrated in Figure 2, our proposed algorithm consists of four components:

- **Implicit Mutual Relations Modeling**: We construct an entity proximity graph to mine the entity pairs with high semantics proximity. In the graph, the semantics proximity can be defined as co-occurrence or similarity between entities in an external unlabeled corpora, rather than the training corpora. Thus, the graph can be constructed in an unsupervised manner. The entities with similar semantics have a similar topological structure in the entity proximity graph. Thus, the implicit mutual relation can be captured by the proximity graph. After vertex embedding, entities will project a low-dimensional space, where entities with similar semantic are closed in the embedding space.
- **Entity Type Embedding**: The entity type is beneficial to filter impossible relations between two entities. For example, entities **Obama** and **Hawaii** are person and location, respectively. The relation between them is absolutely not *childOf*. Thus, we first collect the types of corresponding entities from Freebase , and then embed them into a low-dimensional space. Then we can calculate a confidence score of each relation for the target entity pair by the entity type embedding. The confidence score of relation r means the probability that there is a relation r between the target entity pair.
- **Piecewise CNN with Sentence-Level Attention**: We use the PCNN to encode each sentence s_i into x_i, then the sentences bag S is encoded into $X = \{x_1, x_2, \cdots, x_n\}$. To mitigate effect from the noisy sentence, a sentence-level attention is employed to focus the high quality sentences.
- **Integrating Implicit Mutual Relation and Entity Type into RE Method**: Finally, we integrate the entity types and implicit mutual relation into existing RE approaches. The implicit mutual relations, entity type embedding, and original RE model can calculate the confidence score of each relation separately. The confidence score means the probability that the target entity pair have the corresponding relation. We unify these confidence scores by a linear model and then get the probability that the target entity pair have the relation r.

A. Implicit Mutual Relations Modeling

There are three stages for implicit mutual relations modeling: (1) we construct an entity proximity graph based on the co-existing times of each entity pair; (2) then the entity representation is learned based on the entity proximity graph; (3) we model the implicit mutual relations by the entity representation. The details are shown as follow:

1) Entity proximity graph construction: The entity proximity graph is a graph that captures the semantic relations of the entities. The entities with similar semantic are proximity in the graph, that means they have a similar topological structure in the graph. For example, as shown in Figure 3 (to illustrate more clear, we have omitted some unimportant points and edges), there are direct edges between entities "Houston" and "Dallas" since they are similar in semantic, where the semantic proximity can be simply evaluated by the number of common neighbors between these two entities in the graph.

To model the implicit mutual relation, we first construct an entity proximity graph based on the Wikipedia corpora. We count the co-occurrence times of each entity pair in Wikipedia corpora, where "co-occurrence" refers to two entities appearing in the same sentence. For example, entities "Obama" and "Hawaii" exist in the same sentence "Obama was born in Honolulu, Hawaii.", then the co-occurrence time of "Obama" and "Hawaii" will increase 1.

Each entity is a vertex in the entity proximity graph. An edge will be formed if the co-occurrence time of an entity pair is up to a pre-defined threshold. Furthermore, we model the entity proximity graph as a weighted graph, where a weight of each edge is computed as follows:

$$w_{i,j} = \frac{\log\left(co_{i,j}\right)}{\log\left(\max_{k,l}\{co_{k,l}\}\right)},$$

where the value of $co_{i,j}$ denotes the co-occurrence times of entity pair (e_i, e_j), and $\max_{k,l}\{co_{k,l}\}$ denotes max co-occurrence times of all entity pairs.

In the weighted graph, two vertices with similar topological structure indicate that the corresponding entities have similar semantics in unlabeled corpora. Thus, once we construct the entity proximity graph, the implicit mutual relation can be preserved in it.

2) Entity embedding learning: A natural question is how to ensemble the proximity graph, i.e., the implicit mutual relations, into a relation extraction framework. Following the state-of-the-art network embedding approach [44], we model the implicit mutual relation of entity pair via learning the vertex embedding in the entity proximity graph.

Our goal is to learn the vertex embedding such that vertices with a similar topological structure in the graph are near neighbors in the low-dimensional space. To preserve the graph structure, we define the *first-order* proximity to capture the observed links in the proximity graph, and define the *second-order* proximity to capture the higher-order proximity between vertices in the proximity graph.

To model the *first-order* proximity, for the edge between entities e_i and e_j, the joint probability between e_i and e_j can be defined as follows:

$$P(e_i, e_j) = \frac{1}{1 + exp(-\mathbf{u}_i^T \cdot \mathbf{u}_j)},$$

where $\mathbf{u}_i \in R^d$ is the vector representation of entity e_i in the d-dimensional space. A superior way to preserve the *first-order* proximity is to minimize the distance between $P(e_i, e_j)$ and

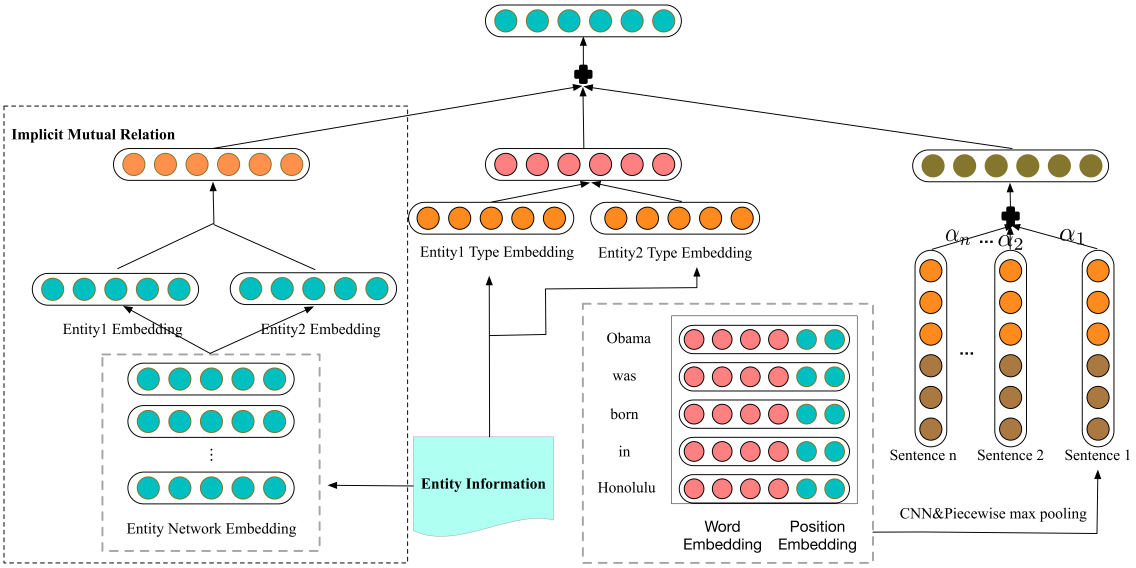


Fig. 2. Overview of our neural relation extraction framework.

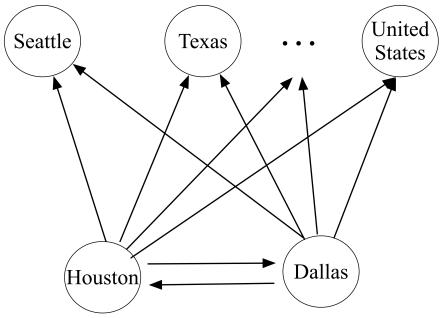


Fig. 3. The similar topological structure of "Houston" and "Dallas".

its empirical probability. When the KL-divergence is chosen to measure this distance, the objective function is as follows:

$$O_1 = -\sum_{(i,j)\in E} w_{ij} \cdot \log P(e_i, e_j).$$

To model the *second-order* proximity, we assume that vertices with many shared neighbors are similar to each other. For each directed edge (e_i, e_j) in the proximity graph, the probability of "context" e_j generated by vertex e_i is defined as

$$P(e_j|e_i) = \frac{\exp\left(\mathbf{u}_j^T \cdot \mathbf{u}_i\right)}{\sum_{k=1}^{|V|} \exp\left(\mathbf{u}_k^T \cdot \mathbf{u}_i\right)},$$

where $|V|$ denotes the amount of vertices.

To preserve the *second-order* proximity, we minimize the distance between $P(e_j|e_i)$ and its empirical probability. Similarly, when the KL-divergence is chosen, the objective function is as follows:

$$O_2 = -\sum_{(i,j)\in E} w_{ij} \cdot \log P(e_j|e_i).$$

In practice, computation of the conditional probability $P(e_j|e_i)$ is extremely expensive. A simple and effective way is to adopt the negative sampling approach mentioned in [44]. Thus, the above objective function can be simplified to

$$O_2 = \log \sigma(\mathbf{u}_j^T \cdot \mathbf{u}_i) + \sum_{i=1}^{K} E_{e_n \sim N(e_i)}[\log \sigma(-\mathbf{u}_n^T \cdot \mathbf{u}_i)],$$

where $\sigma(x) = 1/(1 + exp(-x))$ is the sigmoid function, and K is the number of negative edges. The first term models the observed links, and the second term models the negative links drawn from the noise distribution.

To embed the vertices in the proximity graph, we preserve both the *first-order* proximity and *second-order* proximity separately, then obtain the embedding vector for a vertex by concatenating corresponding embedding vectors learned from the two models.

3) Implicit mutual relation: The vertex embedding vector models the semantic information of an entity. The semantically similar entities, therefore, have close embedding vectors in the embedded space. Thus, we can represent the implicit mutual relation of entities with the entity embedding. The implicit relation between entities e_i and e_j can be represented as follows:

$$MR_{i,j} = \mathbf{U}_j - \mathbf{U}_i,$$

where $\mathbf{U}_i$ is the embedding vector of entity e_i.

B. Entity Type Embedding

In intuition, entity types are beneficial to predict the relation between entities. For example, */people/person/place_of_birth* is the relation between *Location* and *Person*, rather than *Person* and *Person*. Existing works [24], [31], [45] have also shown that entity type information plays a positive role in relation extraction.

Instances in distance supervision learning are based on the sentences aligned to the knowledge graph, where the entity

type information is readily available. Our model uses the entity types defined in FIGER [45], which defines 112 fine-grained entity types. To avoid over-parameterization, our model only employs 38 coarse entity types which form the first hierarchy in FIGER. Each entity type is embedded into k_t dimensional space to get the embedding vector of an entity type. When an entity has multiple types, we take the average over the embedding vectors.

We concatenate the embedding of the types for the target entity pair (e_i, e_j) as follows:

$$T_{i,j} = Concat(Type_i, Type_j),$$

where $Type_i$ is the embedding of type for entity e_i.

C. Piecewise CNN with Sentence-Level Attention

The third component of our approach adopts the sentence-level attention to choose high-quality sentences to training our approach. This component consists of three indispensable steps:

(1) **Sentence Embedding**: Each sentence s_i in a training sentences bag $S = \{s_1, s_2, \cdots, s_n\}$ should be represented by word embedding and relative position embedding. Relative position means the relative position of all words in the sentence to the target entities.

(2) **Sentence Encoding**: As the previous works([19], [13]) shown, the convolutional neural networks with piecewise max pooling (PCNN) is a fast and effective way to encode the sentence. Consequently, we get the encoding of each sentence via using PCNN.

(3) **Sentence-Level Attention**: The distant supervision learning is suffered from noisy labels, i.e., not all sentences in a bag can express the relation for the targeted entity pair. To address this issue, we utilize the sentence-level attention to mitigate effects from the noise sentence. For the encode of each sentence bag, the model gives each sentence in the bag a score according to the quality of this sentence. The encoding of the ith sentence bag can be represented as follows:

$$X_{bag_i} = \sum_{j \in bag_i} \alpha_j x_j,$$

where the X_{bag_i} denotes the bag formed by all training sentences of ith entity pair. The score α_j for sentence j is calculated by the selective sentence attention. It's defined as:

$$\alpha_j = \frac{\exp(q_j)}{\sum_k \exp(q_k)},$$

where q_j is a query-based function which scores how well the sentence j and the predict relation r matches. We use the bi-linear function to calculate the scores:

$$q_j = \mathbf{x}_j \mathbf{A} \mathbf{r},$$

where $\mathbf{A}$ is a weighted diagonal matrix, and $\mathbf{r}$ is the query vector associated with relation r.

D. Combination of Entity Information and RE Method

The implicit mutual relation is a semantic relation between entity pairs. Given the targeted relation set $\{r_1, r_2, \cdots, r_m\}$, the entity pairs with similar implicit mutual relation possibly have the same relation in the relations set. Therefore, we can infer the confidence that the target entity pair has relation r_i via using the implicit mutual relation. We use a fully connected layer with a Softmax activation to calculate the confidence score for each relation. For a target entity pair (e_i, e_j), the confidence inferred from the implicit mutual relation is:

$$C_{MR_{i,j}} = Softmax(W_{MR} MR_{i,j} + b_{MR}),$$

where the W_{MR} and b_{MR} are the parameters of the fully connected layer.

Meanwhile, the entity type information can also give a confidence score to r_i according to the entity type constraints of a relation. We concatenate the type embedding of the target entity pair and then use a fully connected layer with a Softmax activation function to calculate the confidence score. As shown below:

$$C_{T_{i,j}} = Softmax(W_T T_{i,j} + b_T),$$

where the W_T and b_T are the parameters of the fully connected layer.

The original RE model can give a primary prediction of the probability of each relation:

$$RE_{i,j} = Softmax(W_{RE} X_{bag_i} + b_{RE}),$$

where the X_{bag_i} is the ith sentence bag which contains all sentences that the target entity pair (e_i, e_j) co-occurrence in. The W_{RE} and b_{RE} are the parameters of the fully connected layer.

Accordingly, we combine these confidence scores with the original relation extraction (RE) model, to achieve a more accurate result. The probability distribution over m relations between entities e_i and e_j can be computed as follows:

$$P(r_{i,j}) = f(w(\alpha C_{MR_{i,j}} + \beta C_{T_{i,j}} + \gamma RE_{i,j}) + b),$$

where $f(x)$ is Softmax function. The α, β and γ are the weight of three components, which can be learned by the RE model itself.

E. Discussion

The implicit mutual relation can flexibility combine with various relation extraction models. We integrate the implicit mutual relation with some CNN-based and RNN-based models. As shown in Section IV-C, these relation extraction models have significantly improved when combined with implicit mutual relations. We think the implicit mutual relations can also have a positive effect on some other advanced methods.

In our solution, the external source we used can easily collect. For the implicit mutual relations, the only external source we used is the unlabeled corpora (i.e., the Wikipedia dump), which can be directly downloaded from the Wikipedia

TABLE II
THE DESCRIPTIONS OF DATASETS NYT AND GDS.

Datasets	NYT (# Relations: 53)		GDS (# Relations 5)	
Item	# sentences	# entity pairs	# sentences	# entity pairs
Training	522,611	281,270	13,161	7,580
Testing	172,448	96,678	5,663	3,247

website. For the entity types, the relation extraction is a subtask for Knowledge Graph (KG) construction, which means the entity types could be obtained from the KG in most cases. Even if the entity types information is missing, using implicit mutual relation alone can also improve the performance, as shown in the section IV-B.

IV. EXPERIMENTS

We conduct comprehensive experiments to evaluate the performance of our proposed approach by comparing with seven competitors and two variants of our approach on two public datasets. Through the empirical study, we aim at addressing the following research questions:

RQ1: How does our proposed approach perform comparing with state-of-the-art relation extraction approaches?

RQ2: Could the implicit mutual relations and entity types improve the performance of existing relation extraction methods, such as GRU, PCNN, and PCNN + ATT, etc?

RQ3: How do the implicit mutual relations affect the relation extraction model?

In addition, we conduct a case study, which visually demonstrates the effect of the implicit mutual relations.

A. Experimental Settings

1) Datasets: We adopt two widely used public datasets to demonstrate the effectiveness of our method and baselines. They are New York Time(NYT) [22] and Google Distant Supervision (GDS) [46] datasets, where the statistical descriptions of them are illustrated in Table II.

- NYT dataset is generated by annotating entities with Stanford NER tool in the New York Times corpus and then aligns with Freebase to get the relation between entities. The training samples are from the corpus of years 2005-2006, and the testing samples are from the corpus of the year 2007. There are 53 different relations including a relation NA which indicates there is no relation between two entities.
- GDS dataset is an extension of the manually annotated data set Google relation extraction corpus. The entities in Google relation extraction corpus are aligned with web documents and then new sentences contain targeted entities are obtained. There are 5 different relations including a relation NA.

2) Evaluation Metrics: Similar to most existing works, we evaluate our model with the held-out metrics, which compare the predicting relation facts from the test sentences with those in Freebase. We report the precision, recall, f1-score, precision at top N prediction (P@N), and AUC (area under the Precision-Recall curve). For different threshold, the precision and recall are different, so we report the precision and recall at the point of max f1-score. In addition, we compute the average score for each metric after running the same experiment five times.

TABLE III
PARAMETER SETTINGS

Symbol	Description	Value
k_e	Embedding vector size	128
k_t	Entity type embedding size	20
l	Window size	3
k	CNN filters number	230
k_p	POS embedding dimension	5
k_w	Word embedding dimension	50
lr	Learning rate	0.3
l	Sentence max length	120
p	Dropout probability	0.5
n	Batch size	160

3) Parameter Settings: In the experiment, we use the grid search to tune the optimal model parameters. The grid search approach is used to select the learning rate λ for stochastic gradient descent optimizer among $\{0.1,0.2,0.3,0.4,0.5\}$, the sliding window size l of CNN among $\{1,2,3,4,5\}$, the number of filters k of CNN among $\{180,200,230,250,300\}$, and the size of entity type embedding k_t among $\{10,15,20,25,30,40\}$. For the entity embedding size, we follow the setting of [44]. In Table III we show the optimal parameters used in the experiments.

4) Baselines: For evaluating our proposed model, we compare with the following baselines:

Mintz [12] is a traditional distant supervision model which utilizes multi-class logistic regression to extract relations between entities.

MultiR [23] utilizes multi-instance learning to combat the noise from distant supervision learning. It introduces a probabilistic graphical model of multi-instance learning which handles overlapping relations.

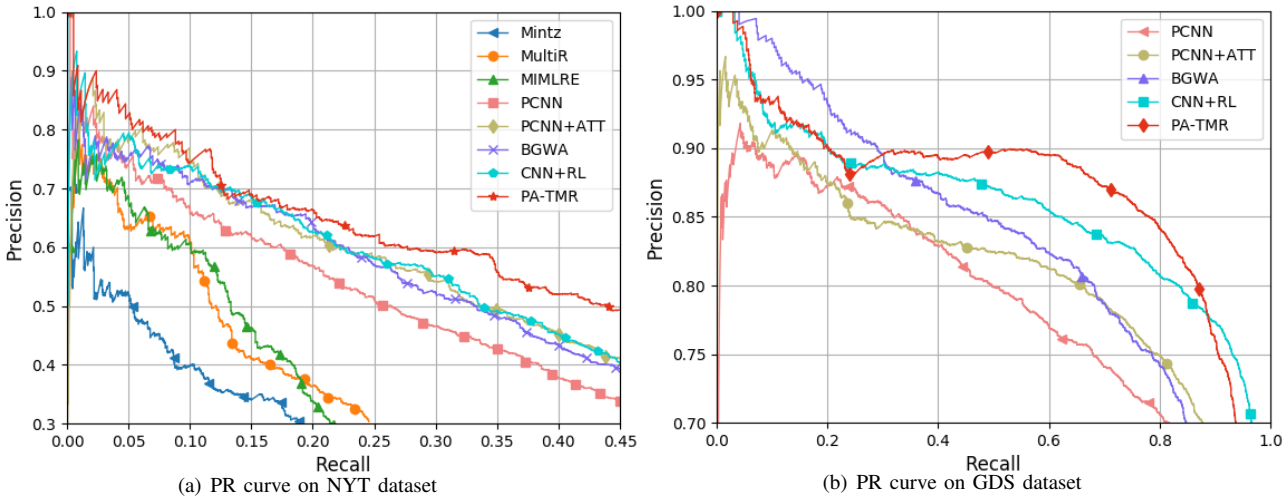
MIMLRE [29] proposes a graphical model which can jointly model the multiple instances and multiple relations.

BGWA [46] is a bidirectional GRU based relation extraction model. It focuses on reducing the noise from distant supervision learning by using a hierarchical attention mechanism.

PCNN [19] is a CNN based relation extraction model which utilizes the piecewise max pooling to replace the single max pooling to capture the structural information between two entities.

PCNN+ATT [13] combines the selective attention over instances with PCNN. The selective attention mechanism is expected to dynamically reduce the weights of those noisy instances, thereby reducing the influence of wrong labeled instances.

CNN+RL [35] contains two modules: an instance selector and a relation classifier. The instance selector chooses high-quality sentences with reinforcement learning. The relation



(a) PR curve on NYT dataset (b) PR curve on GDS dataset

Fig. 4. The Precicion-Recall curve of different algorithms on NYT and GDS datasets

TABLE IV
PERFORMANCE COMPARISON

Dataset	Method	AUC	Precision	Recall	F1-Score	P@100	P@200
NYT	PCNN	0.3296	0.3830	0.4020	0.3923	0.77	0.72
	PCNN+ATT	0.3424	0.3588	0.4564	0.4018	0.75	0.75
	BGWA	0.3670	0.3994	0.4451	0.4210	0.76	0.74
	CNN+RL	0.3735	0.4201	0.4389	0.4293	0.79	0.73
	PA-T	0.3572	0.3779	0.4586	0.4143	0.78	0.72
	PA-MR	0.3635	0.4091	0.4410	0.4244	0.79	0.78
	PA-TMR	**0.3939**	**0.4320**	**0.4615**	**0.4463**	**0.83**	**0.79**
GDS	PCNN	0.7798	0.6804	0.8673	0.7626	0.88	0.90
	PCNN+ATT	0.8034	0.7250	0.8474	0.7814	0.94	0.93
	BGWA	0.8148	0.7725	0.7162	0.8385	0.99	0.98
	CNN+RL	0.8554	0.7680	**0.9132**	0.8343	1.0	0.96
	PA-T	0.8512	0.7925	0.8969	0.8414	0.96	0.94
	PA-MR	0.8571	0.8011	0.8947	**0.8453**	0.97	0.94
	PA-TMR	**0.8646**	**0.8058**	0.8641	0.8339	**1.0**	**0.98**

classifier makes a prediction by the chosen sentences and provides rewards to the instance selector.

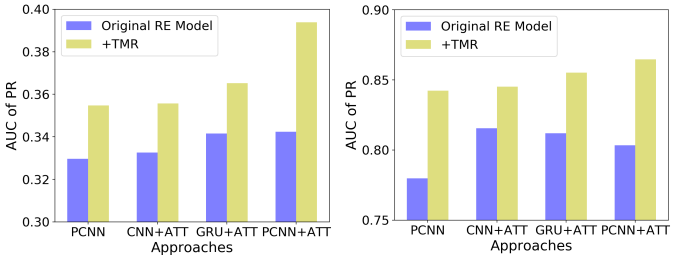
Based on the state-of-the-art relation extraction approach, PCNN+ATT, **PA-TMR** is our proposed approach which integrates entity types and implicit mutual relations into PCNN+ATT approach. In addition, we propose two variants **PA-T** and **PA-MR** which only adopt entity type and implicit mutual relation to improve PCNN+ATT approach, respectively.

*B. Performance Comparison (**RQ1**)*

To verify the effectiveness of our model, we compare our PA-TMR model with baselines on both NYT and GDS datasets as demonstrated in Figure 4. We use the results reported in Lin et al. [13] for the performances of non-neural baselines Mintz [12], MultiR [23] and MIMLRE [29] on NYT dataset. As shown in Figure 4(a), all the non-neural baselines obviously worse than the neural baselines, so we only report the results of neural baselines on GDS dataset. As illustrated in Table IV and Figure 4, we have the following key observations:

- The performance of PCNN is worse than the other neural models. This is due to the factor that the PCNN model does not improve to alleviate the impact from the noisy training sentences, while other neural baselines and our PA-TMR method utilize some techniques, such as reinforcement learning or attention mechanism, to deal with the problem of noisy training sentences. Meanwhile, it reveals the practical necessity to deal with the problem of noisy training sentences in our PA-TMR method.
- Our PA-TMR model not only outperforms all the neural baselines significantly, but also has more obvious advantage when the recall increases as demonstrated in Figure 4. This is due the factors that: (1) all the neural baselines only employ the training corpus to extract relations; (2) the noisy training sentences in distant supervision corpora exacerbates insufficient training problem for the RE models. However, PA-TMR combines the implicit mutual relations and entity types to improve the neural relation extraction. This points to the positive effect of integrating both the implicit mutual relations and entity types into the RE model.

- Comparing to the variants of our PA-TMR method, both PA-T and PA-MR outperform the basic model PCNN+ATT. This improvement illustrates that both the implicit mutual relations and entity types have the positive effect on extracting relations again. Furthermore, PA-TMR achieves the best performance compared to its variants. This sheds the light on the benefit of the interaction of the implicit mutual relations and entity types.



(a) NYT dataset (b) GDS dataset

Fig. 5. The flexibility of our proposed neural RE framework and the improvement of the implicit mutual relations and entity types

C. Flexibility of Our Method (*RQ2*)

To illustrate the flexibility of our PA-TMR method, we corporate the components of implicit mutual relations and entity types into the other neural relation extraction approaches, such as GRU based model with sentence level attention (GRU+ATT), CNN + ATT [13], PCNN [19], and PCNN + ATT [13]. As elaborated in Figure 5, we have the following key observations:

- Comparing to the original models, each improved model achieves 2%-7% improvement by combining the implicit mutual relations, entity types, and distant supervision training corpora. The better performance of the improved models is twofold: (1) all entity pairs with the similar semantic are helpful to extract relation for the target entity pair; (2) the implicit mutual relations can further alleviate the impact from the noisy data in the training corpora. As such, it reveals that only distant supervision training corpora is insufficient for predicting the relation of the target entity pair.
- The original RE models are CNN-based (CNN + ATT, PCNN, PCNN + ATT) or RNN-based (GRU+ATT) approaches. The experimental result illustrates that the basic CNN-based and RNN-based models can achieve significant improvement via only integrating our implicit mutual relations into them without any modification of original approaches. This sheds light on the flexibility of using our proposed implicit mutual relations. Meanwhile, it indicates that the implicit mutual relations can be integrated into most of neural relation extraction methods easily.

D. The Effect of Implicit Mutual Relations (*RQ3*)

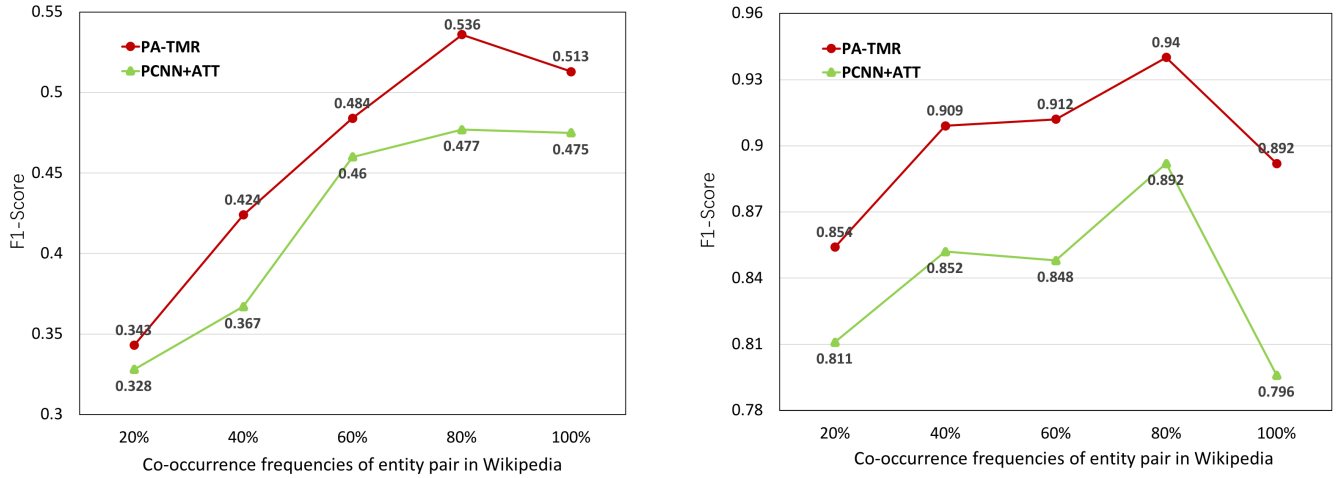
To illustrate the effectiveness of integrating the implicit mutual relations, we first evaluate the performance of our PA-TMR method with different co-occurrence frequencies in unlabeled corpora as illustrated in Figure 6, from which we can find the positive effect of implicit mutual relations of entity pairs with different co-occurrences frequencies in the unlabeled corpora. Then we demonstrate the performance of PA-TMR considering entity pairs with infrequent training instances in the training corpora as shown in Figure 7, which verifies the positive effect of implicit mutual relations for infrequent entity pairs.

1) Improvement from implicit mutual relations: As illustrated in Figure 6, we sort the entity pairs by their co-occurrence frequencies in unlabeled corpora (Wikipedia) and then evaluate the performance for the entity pairs with different co-occurrence frequencies, where the x-axis denotes the quantile of co-occurrence frequencies of entity pairs in Wikipedia, and the y-axis denotes the corresponding F1-score. We have the following key observations:

- As the co-occurrence frequencies of entity pairs increase, the F1-score demonstrates an upwards synchronous trend. It reveals that no matter frequent or infrequent co-occurrences of entity pairs in the unlabeled corpora are helpful for improving the performance of our PA-TMR model. This points to the positive effect of all implicit mutual relations collected from the unlabeled corpora. Meanwhile, the implicit mutual relations, which capture the semantic information of both the target entity pair and the entity pairs with similar semantic, contributes to predict relations for the target entity pair;
- The improvement on the small dataset GDS is much larger than that on NYT dataset. This is due to the factor that: (1) we insufficiently train the original RE model in the smaller dataset; (2) noisy data in a smaller training dataset exacerbates the inadequate issue of training process by utilizing the attention mechanism. The better improvement illustrates that the implicit mutual relations can alleviate the negative impact of insufficient training corpora.

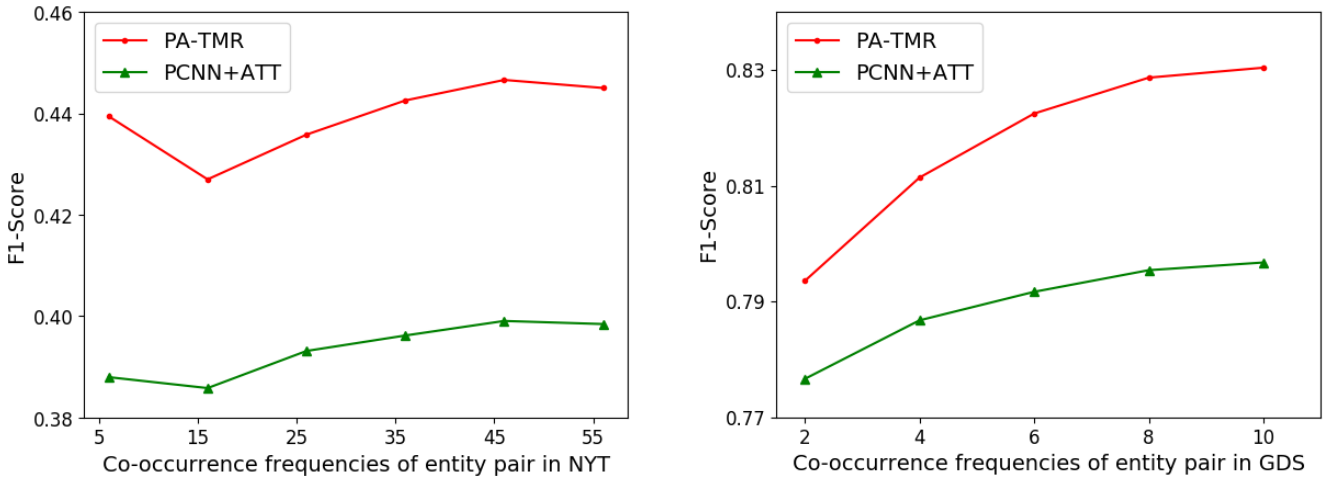
2) The effect on inadequate training sentences: As illustrated in Figure 7, we evaluate the impact of inadequate training sentences, where the x-axis denotes the # training sentences in the distant supervision training corpora, and the y-axis denotes the F1-score of relation extraction for the entity pairs with fixed number of training sentences. We have the following key observations:

- The performance of original PCNN + ATT increases as an entity pair has more training sentences in the distant supervision training corpora. It reveals that inadequate training sentences have negative impact on extracting relations.
- Our PA-TMR method outperforms the PCNN+ATT for extracting relations for the entity pairs with inadequate training sentences significantly. This is due to the factor



(a) NYT dataset (b) GDS dataset

Fig. 6. The f1-score of the test sets with different co-occurrence frequencies of entity pairs.



(a) NYT dataset (b) GDS dataset

Fig. 7. The f1-score of the entity pairs with the different co-occurrence frequencies in original dataset.

that our mined implicit mutual relations contribute to predict the relations of entity pairs with inadequate training sentences.

E. Case Study

In the above experiment, we have identified the effect of implicit mutual relation of entity pairs in extracting relations. It is a natural question that how the improving mechanism of the implicit mutual relations works in the extracting process. Note that the implicit mutual relation is represented as the entity embedding learned from the entity proximity graph. Therefore, we conduct a case study to demonstrate the meanings of the implicit mutual relation after entity embedding.

As shown in Figure 8[1], the embedding vectors of entities are projected into 3D space. We show the nearest entities of **Seattle** and **University of Washington** in the figure. We can observe that most of the nearest entities of **Seattle** are cities in the USA, and most of the nearest entities of **University of Washington** are universities. This suggests that the entities with similar semantic would be closed in the embedding space. The top 10 nearest entities of Seattle and University of Washington are shown in table V. However, there are some entities whose semantics are not close to the target entity pair, such as "San Gabriel Valley". Therefore, in future work, we can adopt more advanced methods to learn the representation of entities to alleviate this problem.

For entity pair (**University of Washington**, **Seattle**), its implicit mutual relation is similar with many other entity pairs, such as (**University of Southern California**, **Los Angeles**) and (**Stanford University**, **California**), etc. Thus, our model tends to correctly predict the relation between "Seattle" and "University of Washington" if we have the high quality

[1]This picture is produced by the Embedding Projector https://projector.tensorflow.org/

TABLE V
THE NEAREST ENTITIES OF SEATTLE AND UNIVERSITY OF WASHINGTON IN EMBEDDING SPACE

Top N	**University of Washington**	**Seattle**
1	University of Florida	New York City
2	University of South California	Washinton
3	Brigham University	California
4	Stanford University	Los Angeles
5	Northwestern University	Texas
6	Ohio State University	Houston
7	University of Michigan	Downtown
8	Bowling Green	San Gabriel Valley
9	Alma	Atlanta
10	University of Kentucky	Cleveland

training instances for entity pairs (**University of Southern California**, **Los Angeles**) and (**Stanford University**, **California**), or the RE approaches correctly predict the relations between entity pairs (**University of Southern California**, **Los Angeles**) and (**Stanford University**, **California**).

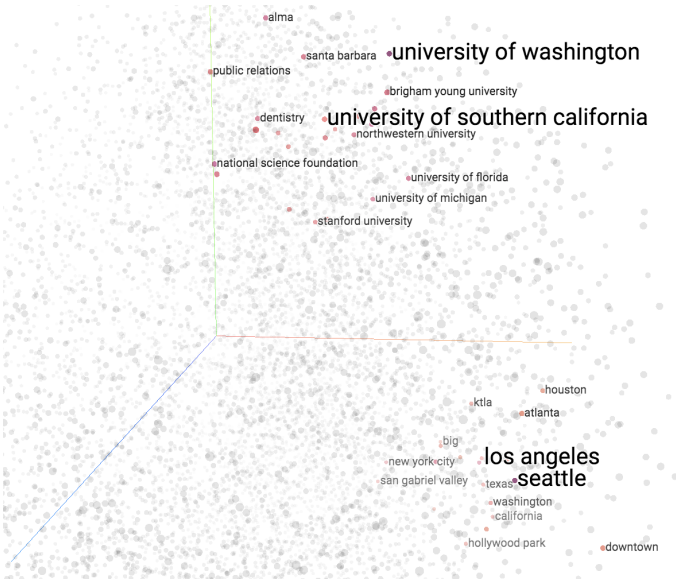


Fig. 8. Visualization of the entity embedding learned from the entity proximity graph.

V. CONCLUSION AND FUTURE WORK

We have presented a unified approach for improving the existing neural relation extraction approaches. In contrast to the existing neural RE models that train the model by only using the distant supervision training corpora, We learn the implicit mutual relations of entity pairs from the unlabeled corpora via embedding the vertices in the entity proximity graph into a low-dimensional space. Meanwhile, our proposed implicit mutual relations are easily and flexibly integrated into existing relation extraction approaches. The experimental results outperform state-of-the-art relation extraction approaches, and manifest that the implicit mutual relations of entity pairs and the entity type information have a positive effect for relation extraction.

In this work, we have only employed the first-order and second-order proximity to capture the implicit mutual relations when we learn the vertex embedding in the entity proximity graph. Thus, it may fail for vertices that have few or even no edges. To address this issue we plan to utilize the graph neural networks (GNNs) [47] or Graph Attention Networks (GATs) [48] to model auxiliary side information, such as numerical features and textual descriptions. In addition, we adopt sentence-level attention to mitigate effects from the noise sentence. As mentioned by Y. Liu et al. [13], the attention mechanism usually alleviates the negative impact of noisy training data. Lastly, we are interested in integrating the other attention mechanism to alleviate the problem.

REFERENCES

[1] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "Dbpedia: A nucleus for a web of open data," in *Proceedings of the 6th International The Semantic Web and 2Nd Asian Conference on Asian Semantic Web Conference*, ser. ISWC'07/ASWC'07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 722–735. [Online]. Available: http://dl.acm.org/citation.cfm?id=1785162.1785216

[2] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: A collaboratively created graph database for structuring human knowledge," in *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '08. New York, NY, USA: ACM, 2008, pp. 1247–1250. [Online]. Available: http://doi.acm.org/10.1145/1376616.1376746

[3] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: A core of semantic knowledge," in *Proceedings of the 16th International Conference on World Wide Web*, ser. WWW '07. New York, NY, USA: ACM, 2007, pp. 697–706. [Online]. Available: http://doi.acm.org/10.1145/1242572.1242667

[4] W. Cui, Y. Xiao, H. Wang, Y. Song, S.-w. Hwang, and W. Wang, "Kbqa: learning question answering over qa corpora and knowledge bases," *Proceedings of the VLDB Endowment*, vol. 10, no. 5, pp. 565–576, 2017.

[5] D. N. Milne, I. H. Witten, and D. M. Nichols, "A knowledge-based search engine powered by wikipedia," in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. ACM, 2007, pp. 445–454.

[6] J. Huang, M. Zhou, and D. Yang, "Extracting chatbot knowledge from online discussion forums." in *IJCAI*, vol. 7, 2007, pp. 423–428.

[7] W. Shen, J. Wang, and J. Han, "Entity linking with a knowledge base: Issues, techniques, and solutions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 2, pp. 443–460, 2014.

[8] N. Bach and S. Badaskar, "A review of relation extraction," *Literature review for Language and Statistics II*, vol. 2, 2007.

[9] N. Kambhatla, "Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations," in *Proceedings of the ACL 2004 on Interactive Poster and Demonstration Sessions*, ser. ACLdemo '04. Stroudsburg, PA, USA: Association for Computational Linguistics, 2004. [Online]. Available: http://dx.doi.org/10.3115/1219044.1219066

[10] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, "Relation classification via convolutional deep neural network." in *Proceeding of the 25th International Conference on Computational Linguistics: Technical Papers*, 2014, pp. 2335–2344.

[11] P. Zhou, W. Shi, J. Tian, Z. Qi, B. Li, H. Hao, and B. Xu, "Attention-based bidirectional long short-term memory networks for relation classification," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2016, pp. 207–212.

[12] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, "Distant supervision for relation extraction without labeled data," in *Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume*. Association for Computational Linguistics., 2009, pp. 1003–1011.

[13] Y. Lin, S. Shen, Z. Liu, H. Luan, and M. Sun, "Neural relation extraction with selective attention over instances." in *Proceeding of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 2124–2133.

[14] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[15] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation." in *Proceding of the 2014 conference on empirical methods in natural language processing*, 2014, pp. 1532–1543.

[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[17] N. Konstantinova, "Review of relation extraction methods: What is new out there?" in *Analysis of Images, Social Networks and Texts*, D. I. Ignatov, M. Y. Khachay, A. Panchenko, N. Konstantinova, and R. E. Yavorsky, Eds. Cham: Springer International Publishing, 2014, pp. 15–28.

[18] R. Socher, B. Huval, C. D. Manning, and A. Y. Ng, "Semantic compositionality through recursive matrix-vector spaces," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, ser. EMNLP-CoNLL '12. Stroudsburg, PA, USA: Association for Computational Linguistics, 2012, pp. 1201–1211. [Online]. Available: http://dl.acm.org/citation.cfm?id=2390948.2391084

[19] D. Zeng, K. Liu, Y. Chen, and J. Zhao, "Distant supervision for relation extraction via piecewise convolutional neural networks." in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 1753–1762.

[20] F. Christopoulou, M. Miwa, and S. Ananiadou, "A walk-based model on entity graphs for relation extraction," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, 2018, pp. 81–88.

[21] M. Miwa and M. Bansal, "End-to-end relation extraction using lstms on sequences and tree structures," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2016, pp. 1105–1116.

[22] S. Riedel, L. Yao, and A. McCallum, "Modeling relations and their mentions without labeled text," in *Proceedings of the 2010 European Conference on Machine Learning and Knowledge Discovery in Databases: Part III*, ser. ECML PKDD'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 148–163. [Online]. Available: http://dl.acm.org/citation.cfm?id=1889788.1889799

[23] R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, and D. S. Weld, "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*, ser. HLT '11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 541–550. [Online]. Available: http://dl.acm.org/citation.cfm?id=2002472.2002541

[24] S. Vashishth, R. Joshi, S. S. Prayaga, C. Bhattacharyya, and P. Talukdar, "Reside: Improving distantly-supervised neural relation extraction using side information," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2018, pp. 1257–1266.

[25] W. Zeng, Y. Lin, Z. Liu, and M. Sun, "Incorporating relation paths in neural relation extraction," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2017, pp. 1768–1777.

[26] G. Ji, K. Liu, S. He, and J. Zhao, "Distant supervision for relation extraction with sentence-level attention and entity descriptions," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[27] C. N. d. Santos, B. Xiang, and B. Zhou, "Classifying relations by ranking with convolutional neural networks," *arXiv preprint arXiv:1504.06580*, 2015.

[28] T. H. Nguyen and R. Grishman, "Relation extraction: Perspective from convolutional neural networks," in *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, 2015, pp. 39–48.

[29] M. Surdeanu, J. Tibshirani, R. Nallapati, and C. D. Manning, "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*, ser. EMNLP-CoNLL '12. Stroudsburg, PA, USA: Association for Computational Linguistics, 2012, pp. 455–465. [Online]. Available: http://dl.acm.org/citation.cfm?id=2390948.2391003

[30] V. Mnih, N. Heess, A. Graves *et al.*, "Recurrent models of visual attention," in *Advances in neural information processing systems*, 2014, pp. 2204–2212.

[31] Y. Yadollah, H. Adel, and H. Schtze., "Noise mitigation for neural entity typing and relation extraction." in *the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1,*. Association for Computational Linguistics, 2017, pp. 1183–1194.

[32] T. Liu, X. Zhang, W. Zhou, and W. Jia, "Neural relation extraction via inner-sentence noise reduction and transfer learning," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2018, pp. 2195–2204. [Online]. Available: http://aclweb.org/anthology/D18-1243

[33] J. Du, J. Han, A. Way, and D. Wan, "Multi-level structured self-attentions for distantly supervised relation extraction," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2018, pp. 2216–2225. [Online]. Available: http://aclweb.org/anthology/D18-1245

[34] L. Wang, Z. Cao, G. De Melo, and Z. Liu, "Relation classification via multi-level attention cnns," 2016.

[35] J. Feng, M. Huang, Z. Li, Y. Yang, and X. Zhu, "Reinforcement learning for relation classification from noisy data," in *Proceeding of The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*, 2018, pp. 5779–5786.

[36] P. Qin, W. Xu, and W. Y. Wang, "Robust distant supervision relation extraction via deep reinforcement learning," in *ACL*, 2018.

[37] Y. Wu, D. Bamman, and S. Russell, "Adversarial training for relation extraction," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 1778–1783.

[38] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[39] P. Qin, W. Xu, and W. Y. Wang, "Dsgan: Generative adversarial training for distant supervision relation extraction," *arXiv preprint arXiv:1805.09929*, 2018.

[40] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[41] Y. Liu, K. Liu, L. Xu, and J. Zhao, "Exploring fine-grained entity type constraints for distantly supervised relation extraction," in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 2014, pp. 2107–2116.

[42] Y. Pinter and J. Eisenstein, "Predicting semantic relations using global graph properties," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2018, pp. 1741–1751.

[43] X. Huang *et al.*, "Attention-based convolutional neural network for semantic relation extraction," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2016, pp. 2526–2536.

[44] J. Tang, M. Qu, M. Wang, and et al., "Line: Large-scale information network embedding," in *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee., 2015, pp. 1067–1077.

[45] X. Ling and D. S. Weld., "Fine-grained entity recognition," in *AAAI'12*. AAAI Press, 2012, pp. 84–100.

[46] S. Jat, S. Khandelwal, and P. Talukdar, "Improving distantly supervised relation extraction using word and entity based attention," *arXiv preprint arXiv:1804.06987*, 2018.

[47] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: going beyond euclidean data," *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 18–42, 2017.

[48] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.