Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 8889075, 16 pages
https://doi.org/10.1155/2021/8889075

Research Article

Distant Supervision for Relation Extraction with Sentence Selection and Interaction Representation

Tiantian Chen,[1] **Nianbin Wang**,[1] **Hongbin Wang**,[1] **and Haomin Zhan**[2]

[1]*College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China*
[2]*Beijing General Institute of Electronic Engineering, Beijing 100854, China*

Correspondence should be addressed to Hongbin Wang; wanghongbin@hrbeu.edu.cn

Received 25 May 2020; Revised 2 December 2020; Accepted 25 January 2021; Published 16 February 2021

Academic Editor: Javier Prieto

Distant supervision (DS) has been widely used for relation extraction (RE), which automatically generates large-scale labeled data. However, there is a wrong labeling problem, which affects the performance of RE. Besides, the existing method suffers from the lack of useful semantic features for some positive training instances. To address the above problems, we propose a novel RE model with sentence selection and interaction representation for distantly supervised RE. First, we propose a pattern method based on the relation trigger words as a sentence selector to filter out noisy sentences to alleviate the wrong labeling problem. After clean instances are obtained, we propose the interaction representation using the word-level attention mechanism-based entity pairs to dynamically increase the weights of the words related to entity pairs, which can provide more useful semantic information for relation prediction. The proposed model outperforms the strongest baseline by 2.61 in F1-score on a widely used dataset, which proves that our model performs significantly better than the state-of-the-art RE systems.

1. Introduction

The relation extraction (RE) task is to identify the relational facts from plain text. It is an important task in natural language processing (NLP) and has been widely used in many intelligent applications such as knowledge graph (KG) construction [1] and question answering (QA) [2].

In recent years, supervised learning methods have achieved great progress in the RE task. However, as with other NLP tasks, most existing supervised RE methods suffer from the lack of high-quality training data, because manually labeled data is time-consuming and labor-intensive.

To solve the above human-labeled data problem, distant supervision (DS) is firstly used by Mintz et al. [3] for RE, which can build training data quickly and automatically by aligning plain texts and knowledge base (KB). DS is based on the following assumption: if two entities have a relation in the triple of KB, then all sentences that contain these two entities will express this relation.

Although DS is an effective method to annotate corpus automatically, its assumption is too strong. When there is only one relation in an entity pair in triples of KB, sentences that do not express the relation in the triple are still forced to be labeled this relation. Thus, DS always suffers from a noisy labeling problem. Figure 1 shows some examples of the alignment between plain texts and KB via DS. For example, the relation /business/company/founders and entity pair (Microsoft, Bill Gates) constitute a triple in KB; the sentences from S1 to S6 containing the entity pair (Microsoft, Bill Gates) will be considered as valid instances for relation /business/company/founder. The first four sentences describe the relation between the entity pair (Microsoft, Bill Gates) as /business/company/founders. However, the sentences S5 and S6 express different relations; they are wrongly labeled as training instances for the relation /business/company/founders. Thus, DS introduces noise into the training data. In addition, the entity pair may have multiple relations in KB. When the triples and the texts are aligned, the DS assumption may also fail, which results in wrong labels. For instance, there are two relations between two entities (New Zealand, Wellington) in Freebase, namely, /location/location/contains and /location/country/capital. According to DS, sentences S7 to S9

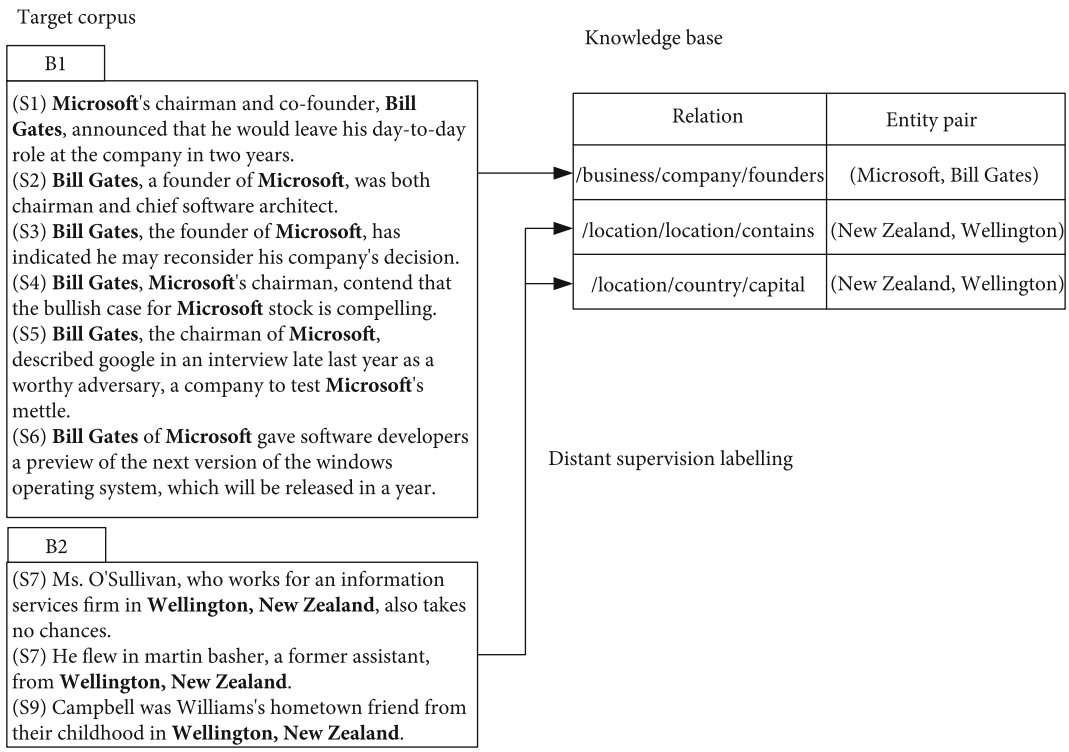

Figure 1: The alignment process of DS between KB and plain texts.

containing the same entity pair (New Zealand, Wellington) are labeled as the above two relations, respectively. But in fact, these three sentences only correctly describe the relation /location/location/contains, and the relation /location/country/capital is the noisy label. Therefore, DS introduces noise data in the dataset regardless of the single relation or multiple relations between the entity pairs in KB. Ru et al. [4] have surveyed the percentage of noisy labeled data introduced by DS in a real corpus which is resulted from a subset of Wikipedia. The average error rate of labeled data is 74.1%, which may seriously affect the performance of RE. The wrong labels generated by DS is a very tricky problem and a main challenge in the RE task. If the noisy labeled data can be removed from the training data, it is promising to greatly improve the performance of DS for RE.

To alleviate the wrong labeling problem, multi-instance learning [5–7] is applied to RE. The multi-instance learning divides sentences containing the same entity pair into a bag, which is labeled as the corresponding relation. Therefore, the training and testing process has proceeded at the bag level. However, traditional feature-based methods [5–7] have typically applied traditional machine learning models and elaborately designed features to training data. Most of these features are extracted directly by NLP tools for RE. Meanwhile, there are inevitable errors in the NLP tools that can lead to error propagation or accumulation. To avoid relying on external tools, some recent works [8–11] proposes to apply deep neural networks to extract features for RE without the NLP tools. These neural network methods automatically extract features for labeled training data obtained by DS, avoiding the dependence on NLP tools and without the need for artificially well-designed features. However, existing extraction methods in DS still face two challenges.

(1) Wrong Label. In multi-instance learning, the bag contains noisy sentences mentioning the same entity pair but possibly not describing the same relation. As a result, the bags contain positive instances and partial noise data. Zeng et al. [8] apply the multi-instance learning strategy and at least one hypothesis to extract relations on training data, which choose a most likely useful sentence to represent the bag. However, when there is no sentence in the bag describing the relation, a sentence is still selected to represent the bag. Lin et al. [10] use sentence-level attention to encode the bag. Although their method has proven to be effective, the harmful noisy sentences are also assigned small but still positive weights, which means the noise effect is not eliminated. Ru et al. [4] propose a semantic Jaccard algorithm to reduce the wrong labels. These methods have effectively alleviated the impact of noisy data on RE tasks, but there are still wrong labeling problem, especially if an entity pair has multiple labels.

(2) Feature Sparsity. Another challenge is the feature sparsity problem, namely, RE suffers from the lack of useful semantic features. Zeng et al. [8] use at least one hypothesis to select only one sentence to represent the bag information for the same entity pair, which will lose a large amount of useful information containing in those neglected sentences. Lin et al. [10] propose sentence-level attention to assign different weights to all sentences in the same entity pair. In the RE task, a single sentence contains less semantic information, and most RE

methods extract global context features of sentences while ignoring the crucial information related to entity pairs.

To handle the first challenge, inspired by Wang et al. [11], we propose a pattern method based on relation trigger word, which is as a sentence selector to filter wrong labeling sentences. Wang et al. [11] assume that each relation in KG has one or more sentence patterns that can describe the meaning of the relation, and they replace the entities contained in the sentence with the type of the entity in KG to generate the sentence pattern. For example, in the sentence "He is next scheduled to perform with the Ornette Coleman quartet in Kongsberg, Norway, on July 6.", their method replaces "in Kongsberg, Norway" with the type "in PLACE, PLACE" to form a sentence pattern "in A, B" which means "B contains A" and indicates the relation "/location/location/contains". Then, this sentence is expressed as "He is next scheduled to perform with the Ornette Coleman quartet in PLACE, PLACE, on July 6." Different from their method, we analyze the structure of the sentence instead of substituting the entity with the type and use the pattern method based on relation trigger word to filter noisy sentences. More specifically, the sentence selector selects the relation trigger words by calculating the semantic similarity between the related phrases in the KB and the sentences containing the entity pairs. Then, the selector uses the pattern method based on the relation trigger words to determine if the sentences are noisy and filter sentences with wrong labels. Ideally, the sentence selector filters out all the wrong label sentences, so that the model can construct a dataset similar to the supervised RE, which is conducive to improve the accuracy of RE. To solve the second challenge, we use the interaction representation between entity pairs and sentences as a supplementary feature for the relation extractor to better reflect the semantic relation between words and entity pairs in the sentence. Specifically, our model uses the word-level attention mechanism-based entity pairs to dynamically increase the weights of the words related to entity pairs and reduce the weights of the trivial words in the sentences. The words related to entity pairs may be related to relations, which may provide more useful information for RE. Ideally, the words related to entity pairs will become the main components in the sentence encoding, which may improve the performance of RE.

The major contributions of our work can be listed as follows.

(1) To solve the wrong labeling problem, we propose a pattern method based on the relation trigger words as a sentence selector to filter out noisy sentences. The selector can extract the relation trigger words according to the semantic similarity between the relation phrase in KB and the sentence in the corpus and choose the high-quality sentences via pattern based on the relation trigger words

(2) To alleviate the feature sparse problem, we present an interaction representation between entity pairs and sentences as a supplementary feature to make full use of the implicit information existing in the sentence

(3) Experiments on real-world datasets indicate that our approach outperforms previous state-of-the-art baselines

2. Related Work

RE is one of the widely studied topics in NLP. To improve the performance of the relation extractor, various supervised learning methods have been proposed, which mainly include Naive Bayes [12], support vector machines [13], and maximum entropy [14]. Although supervised methods demonstrate superior performance in RE tasks, these methods rely overly on human-annotated training data. These data annotated are expensive and inefficient. In addition, the quantity of these data is limited.

To address the above problem, DS is proposed by Mintz et al. [3] to generate a large corpus without expensive manual annotation. However, DS introduces noise data, and these noises will seriously hurt the performance of RE. To reduce the influence of noise data, Riedel et al. [5] use multi-instance learning and expressed at least once assumption for distantly supervised RE. Hoffmann et al. [6] propose a probabilistic graphical model to deal with overlapping relations in the RE task. Surdeanu et al. [7] learn a Bayesian framework by expectation maximization (EM) algorithm.

Recently, neural network models have been successfully applied to RE. Zeng et al. [8] propose the piecewise convolutional neural network (PCNN) that segments the sentence according to the position of the entity pair, improving the performance of distantly supervised RE. Wen et al. [9] believe that PCNN lacks the consideration of the impacts of entity pairs and the sentence context on word encoding and does not distinguish the different contributions of the three segments in PCNN to relation classification, so they introduce a novel gated PCNN for distantly supervised RE. Lin et al. [10] first present a sentence-level attention mechanism to assign higher weights to all the valid sentences in the bag and achieve amazing results. Zeng et al. [15] introduce a path-based neural extraction model to encode the relational semantic information from both direct sentences and inference chains that can be built between two target entities via intermediate entities. Motivated by the hypertext-induced topic search (HITS) [16] algorithm, and selecting cluster centroids method such as K-means, latent semantic analysis (LSA) [17], or nonnegative matrix factorization (NMF) [18], Phi et al. [19] formulate wrong label reduction tasks as ranking problems according to different ranking criteria. He et al. [20] divide the original classification task into subtasks in different levels and construct a tree-like categorization structure. With the tree-like structure, unlabelled relation sentences are progressively categorized along the path from the root node to the leaf node. Ru et al. [4] use a semantic Jaccard similarity algorithm to select a core dependency phrase to represent the sentence to alleviate the noise data. Qu et al. [21] inspired by TransE [22] and KG completion [23] use the approximate representation of the relation vector to calculate the attention of each word in the sentence. Zhou et al. [24] present a novel hierarchical selective attention model for RE, which uses coarse sentence-level attention

to select the most relevant sentences and word-level attention to obtain sentence representations. Zhao et al. [25] also design a hierarchical-attention mechanism, which is aimed at selecting the most informative features for RE. Sun et al. [26] introduce a multihead self-attention network to learn the sentence representation without any convolutional and recurrent operations. Vashishth et al. [27] present a novel RE method based on graph convolution networks, which makes use of relevant side information, such as entity type and relation alias from KB, for improving distantly supervised RE. To effectively alleviate class imbalance, Ye and Luo [28] present a general ranking-based multilabel learning framework combined with the convolutional neural network (CNN). Mitra et al. [29] propose a multiview-based deep neural network model, which combines CNN and Bidirectional Long Short Term Memory (Bi-LSTM) network along with a multilayer perceptron (MLP). Shi et al. [30] propose an advanced graph neural network, which assigns higher weights to those direct neighbor words that contribute more to relation prediction through breadth exploration. Ouyang et al. [31] use graph attention networks to encode syntactic features, which obtain the important semantic information of related words in each sentence. Phi et al. [32] combine a bidirectional gated recurrent unit (BiGRU) model with a form of hierarchical attention that enhances the performance of the distantly supervised RE task. To get better sentence context representation, Jat et al. [33] propose two word-level attention models for distantly supervised RE, viz., a BiGRU-based word-level attention model and an entity-centric attention model. Geng et al. [34] employ bidirectional tree-structured long short-term memory (LSTM) to extract structural features based on the dependency tree in the sentence. Ye et al. [35] present a unified framework to integrate relation constraints with the neural network by introducing constraint loss.

In addition to the methods above, reinforcement learning (RL) has been successfully applied to RE tasks to improve extracting performance. Feng et al. [36] and Zeng et al. [37] introduce RL to select high-quality sentences. Qin et al. [38] employ RL to redistribute wrong-labeled instances into the negative set. Sun et al. [39] propose an RL-based bag-level label denoising model, which applies the policy network to correct wrong labels. Chen et al. [40] present a sentence-level label denoising model based on RL to solve the noisy labeling problem.

These methods have effectively reduced the impact of wrongly annotated data in the RE task, but noisy sentences are not completely removed. Meanwhile, most methods have the feature sparsity problem. Thus, we propose a distantly supervised RE method with sentence selection and interaction representation to filter out noisy sentences and to extract more useful information from sentences. The differences between our method and previous methods are as follows: Lin et al. [10] use the sentence-level attention method to assign more weight to useful sentences and effectively alleviate the wrong labeling problem, but they also assign small weights to harmful noisy sentences. Therefore, the noise problem still exists. Feng et al. [36], Zeng et al. [37], Qin et al. [38], and Sun et al. [39] use the policy gradient method to solve the wrong labeling problem. Chen et al. [40] employ Deep Q Network (DQN) to reduce the noisy labels. Feng et al. [36] and Zeng et al. [37] filter noise sentences. Qin et al. [38] redistribute false-positive samples into negative examples. Sun et al. [39] correct the bag-level noisy labels, but the bags still contain noisy sentences. Chen et al. [40] aim to filter sentence-level noise labels. In addition, these methods mentioned above all use the typical neural network model PCNN or CNN to obtain the feature representation of the sentence, ignoring the implicit information of the sentence and the correlation degree between the entity pair and the different words in the sentence. Unlike them, we use the related phrases in the KB as prior knowledge to obtain relation trigger words and propose a pattern method based on the relation trigger words as a sentence selector to filter out noisy sentences. Moreover, to obtain more sentence feature information, we propose the interaction representation using the word-level attention mechanism-based entity pairs to allot larger attention weights to those words related to entity pairs and improve the performance of RE.

3. Methodology

As shown in Figure 2, based on the original labeled data generated by DS, we decompose the RE task into two subproblems in our work: sentence selecting and relation extracting. For the sentence selecting problem, we construct a sentence selector to filter out sentences annotated with wrong labels by pattern method based on the relation trigger words. The selector extracts the relation trigger words via the semantic similarity between the relation phrase in KB and the sentence in the dataset of DS and chooses the high-quality sentences via pattern based on the relation trigger words. For the relation extracting problem, we present the interaction representation between entity pairs and sentences, which fully considers the useful information implied by sentences and entity pairs. Then, we use PCNN to automatically learn sentence features based on high-quality sentences and concatenate these features and interaction representation as the entire sentence representation. Afterward, we employ the existing sentence-level attention model to acquire the bag representation. Finally, the bag representation is fed into the softmax classifier to predict the relation.

3.1. Problem Definition

3.1.1. Distantly Supervised Data. The distantly supervised data (h, t, r, s) is generated by aligning the triple (h, t, r) composed of the head entity h, tail entity t, and relation r in the existing KB with the plain text s. For example, the entity pair (Bill Gates, Microsoft) in Figure 1 has a relation /business/company/founders in freebase, and sentence S1 containing these two entities is labeled as relation /business/company/founders.

3.1.2. Sentence Selecting. Given a set of data $X = \{(h_1, t_1, r_1, s_1), (h_2, t_2, r_2, s_2), \cdots\cdots, (h_n, t_n, r_n, s_n)\}$, target relation sets $R = \{r_1, r_2, \cdots\cdots, r_l\}$, and pattern sets $P = \{p_1, p_2, \cdots\cdots, p_m\}$, sentence selection is aimed at selecting the correct sentences by pattern method based on the relation trigger words. A

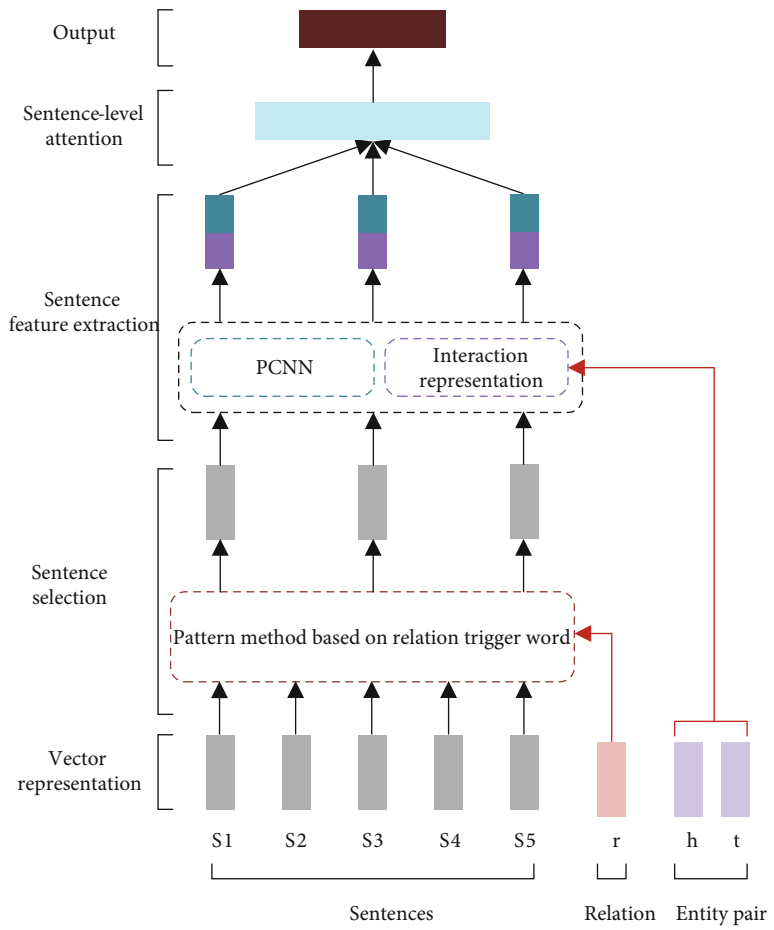

Figure 2: The overall architecture of our method for the RE task.

relation r can correspond to multiple patterns p, where $r \in R$ and $p \in P$. The pattern p is denoted as a triple $(\text{type}_1, \text{trigger}, \text{type}_2)$, where type_1 is the type of head entity h, type_2 is the type of tail entity t, and trigger is the relation trigger words that can represent the target relation in any entity pairs. Instance-pattern p' is expressed as the triple $(h, \text{trigger}, t)$, which is the entity types in pattern triple are replaced with the specific entity pair. For example, the sentence "He is next scheduled to perform with the Ornette Coleman quartet in Kongsberg, Norway, on July 6." containing the entity pair (Kongsberg, Norway) in Table 1 has a relation label /location/location/contains. It is obvious that the pattern p is (LOCATION, in, LOCATION), and the instance-pattern p' is (Kongsberg, in, Norway) in this sentence.

3.1.3. Relation Extracting. In multi-instance learning, sentences with the same entity pair compose a bag, expressed as $(h, t, \{s_1, s_2, \cdots \cdots s_c\})$; RE aims to identify the relation r for the entity pair (h, t) in the bag.

3.2. Sentence Selection

3.2.1. Relation Trigger Words. To better distinguish between the wrong and the correct label, we use the cosine similarity algorithm to calculate the semantic similarity between words and related phrases. The word embedding is used to represent all the words in sentences and relational phrases of KB. Then, we calculate the cosine similarity of word vectors between the relation phrase and the sentence. The maximum value of the cosine similarity is the semantic similarity between the sentence and the related phrase. The threshold we set is used to filter sentences with low similarity, and the remaining sentences are sent as input to the relation extractor.

(1) Word Embedding. Word embedding is a distributed representation of words that maps words in sentences and related phrases to real-valued vectors. For each sentence, the vector of the word W is expressed as WE_i, the sentence consisting of m words is represented as $s = [\text{WE}_1; \text{WE}_2; \cdots \cdots; \text{WE}_m] \in \mathbb{R}^{m \times d_w}$, where m is the number of words in a sentence, and d_w is the dimension of word vector. Given a relation phrase containing n words, the word vector in the related phrases is represented as VE_j. Thus, we use $r = [\text{VE}_1; \text{VE}_2; \cdots \cdots; \text{VE}_n] \in \mathbb{R}^{n \times d_w}$ to express the relation encoding, where n is the number of words in a relation phrase.

We calculate the cosine similarity value for each word vector between the sentence and the relation phrase:

Table 1: The pattern of specific relation for the instances.

Sentences	Labels (relations)	Patterns
He is next scheduled to perform with the Ornette Coleman quartet in **Kongsberg**, **Norway**, on July 6.	/Location/location/contains	(LOCATION, in, LOCATION)
..., Said Mr.Cho, 25, who wanted to **Seoul**, **South Korea**, and educated at a boarding school in Scotland.	/Location/location/contains	(LOCATION, to, LOCATION)
She said the state's division of special revenue was investigating the incident, which took place at the studios of **Wtic** television in **Hartford**, where the drawing is televised.	/Broadcast/content/location	(BROADCAST, in, LOCATION)
He was **George Mcgovern** of **South Dakota**–not Frank church of Idaho, who was involved in other antiwar legislation.	/People/person/nationality	(PEOPLE, in, LOCATION)
We definitely needed to make a change, "Gian Giacomo Ferraris, the chief executive of Jil Sander, said from **Milan**, where **Prada** is based.	/Business/business_location/parent_company	(LOCATION, where is based, COMPANY)
Monsanto, based in **St._Louis**, licenses its biotechnology traits to seed companies, including Delta, which incorporate them in their own varieties.	/Business/company/place_founded	(COMPANY, based in, LOCATION)
Noting that **Charles Darwin** is buried in **Westminster Abbey**, Dr. Barrow said that in contrast with the so-called culture wars in America, science and religion had long coexisted peaceably in England. "	/People/place_of_interment/interred_here	(PEOPLE, buried in, LOCATION)
President **Franklin D. Roosevelt** was born in **Hyde Park** in Dutchess county and often resided there during his presidency, inspiring the country to live without fear through the great depression and world war ii.	/People/person/place_of_birth	(PEOPLE, born in, LOCATION)

$$\mathrm{sim}\left(W, \mathrm{WE}_i, \mathrm{VE}_j\right) = \frac{\mathrm{WE}_i \cdot \mathrm{VE}_j}{\|\mathrm{WE}_i\| \times \left\|\mathrm{VE}_j\right\|}. \tag{1}$$

The semantic similarity scores between a relation phrase and a sentence are defined as follows:

$$\mathrm{sim}(s, r) = \max \mathrm{sim}(W, \mathrm{WE}, \mathrm{VE}), \tag{2}$$

$$\mathrm{WORD} = \arg\max_{W} \mathrm{sim}(W, \mathrm{WE}, \mathrm{VE}), \tag{3}$$

where WORD represents the trigger word.

Given a similarity threshold δ, if the semantic similarity score between the sentence and the relation is not less than the threshold δ, then the sentence has the relation trigger word; otherwise, the sentence has no relation trigger word.

$$\mathrm{trigger} = \begin{cases} \mathrm{WORD} & \mathrm{sim}(s, r) \geq \delta, \\ \varnothing & \mathrm{sim}(s, r) < \delta, \end{cases} \tag{4}$$

where $\varnothing$ represents no relation trigger word.

3.2.2. Pattern Based on the Relation Trigger Words. The training data generated by the DS still have a lot of noise, which makes the effect of the RE task unsatisfactory. If the wrong labels can be removed from the training data, the performance of the RE can be greatly improved. Wang et al. [41] propose a label-free DS method for RE via KG embedding. They make no use of the relation labels under distantly supervised datasets, but only use the prior knowledge derived from the KG to supervise the extractor learning directly and softly. They assume that each relation in the KG has one or more sentence patterns that can describe the meaning of the relation. Their model achieves good results, which fully proves that each relation has one or more sentence patterns describing its meaning. Therefore, we believe that the pattern method based on the relation trigger words can also effectively reduce wrong labels for RE tasks.

In the training data obtained by DS, when there are multiple relations between two entities, the assumption of DS may fail. Thus, we first use the pattern method based on the relation trigger words to choose the correct sentences. Target relation sets R, pattern sets P, and sentence sets S are given, where $r \in R$,$p \in P$, and $s \in S$. A relation label set $\mathrm{Relation}(h, t, s) = \{r_1, r_2, \cdots \cdots, r_n\}$ denotes the set of relation label in the sentence s containing the same entity pairs(h, t), where the correlation between sentences and relations in DS can be represented by the bipartite graph in Figure 3(a). A relation r can correspond to 0 or multiple patterns p, and$\mathrm{Pattern}(r) = \{p_1, p_2, \cdots \cdots, p_n\}$ represents the pattern set corresponding to the relation, where $\mathrm{Pattern}(r) \subseteq P$. In the paper, we only consider that a pattern can express at most one relation, so that the relation r corresponding to the pattern p is represented by $\mathrm{Relation}'(p) = r$. The relations and patterns in Figure 3(b) are one-to-many correspondences. Because a pattern p can match multiple sentences, and a sentence s can correspond to multiple patterns, patterns and sentences have many-to-many correspondences in Figure 3(c). Similarly, the pattern p and the instance-pattern p' have one-to-many correspondences in Figure 3(d), and the instance-pattern p' and the sentence $s$with the same entity pair have many-to-many correspondences in Figure 3(e).

To alleviate the noisy data, the pattern method based on the relation trigger words is used to select high-quality sentences. The main idea of the pattern method: if the relation corresponding to the pattern is in the label set of the sentence and the instance-pattern appears in the sentence, we determine that the relation corresponding to the pattern is the

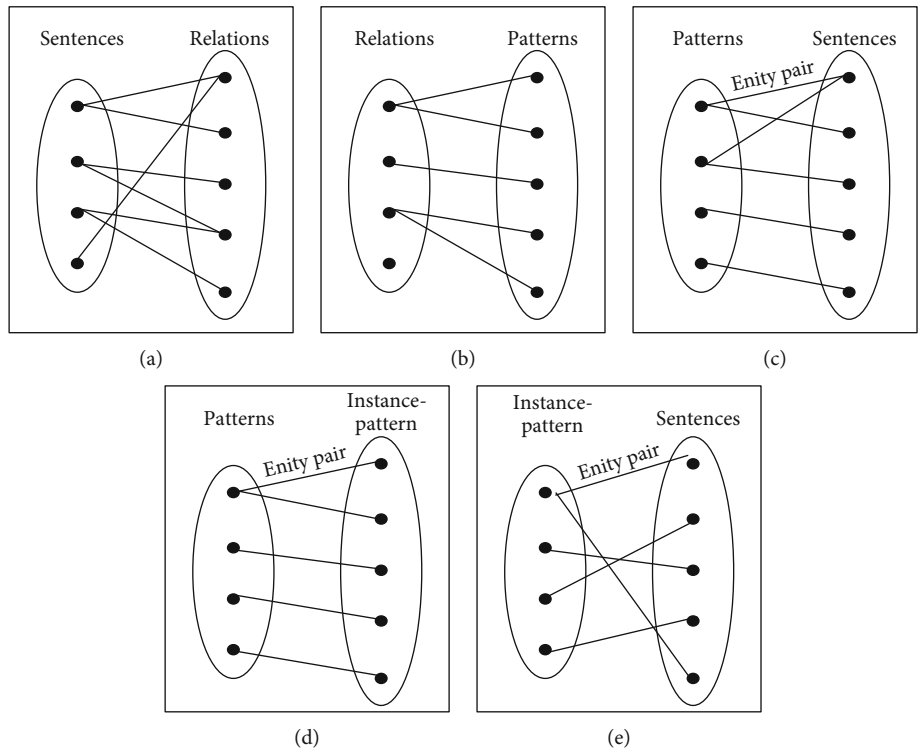

Figure 3: Graph representations of the correlation among sentences, relations, patterns, and instance-pattern.

correct label for the sentence; otherwise, the label is wrong. It is described as

$$M\left(p', s\right) = \text{Include}\left(p', s\right) \wedge R(p, s), \tag{5}$$

where $\text{Include}(p', s)$ is used to determine whether the instance-pattern is in the sentence. $R(p, s)$ is used to determine whether the relation corresponding to the pattern is in the label set of the sentence. When the pattern and the sentence, the instance-pattern and the sentence are successfully matched at the same time, the relation label of the sentence is considered to be correct, and $M(p', s)$ is 1, otherwise, it is 0.

The matching result of the pattern and the sentence is expressed as

$$R(p, s) = \begin{cases} 1, & \text{Relation}'(p) \in \text{Relation}(h, t, s), \\ 0, & \text{otherwise}, \end{cases} \tag{6}$$

where $\text{Relation}(h, t, s)$ is the relation label set of the sentence s that contains the same entity pair (h, t). If the label set of the sentence includes the relation corresponding to the pattern, it is considered that the current sentence s and pattern p are matchable. When the pattern matches the sentence, $R(p, s)$ is 1, otherwise, $R(p, s)$ is 0.

After the matching result of the pattern and the sentence is obtained, an entity pair in the sentence replace the entity types of pattern to generate the instance-pattern. Then, we define the matching formula between the instance-pattern, and the sentence is defined as

$$\text{Include}\left(p', s\right) = \begin{cases} 1, & \text{if } p' \text{ in } s, \\ 0, & \text{otherwise}. \end{cases} \tag{7}$$

If the instance-pattern appears in the sentence, the instance-pattern p' is considered to match the sentence s, which $\text{Include}(p', s)$ is 1, otherwise, $\text{Include}(p', s)$ is 0.

The pattern method can select high-quality sentences. For example, the relation trigger word "founded" of the sentence S1 in Figure 1 and the related word "founders" have a strong semantic relation. Such sentences are easily selected. However, the semantic relation between the relation trigger word and the relation label in some sentences is relatively weak. As shown in Table 1, the sentence "He is next scheduled to perform with the Ornette Coleman quartet in Kongsberg, Norway, on July 6." has a relation trigger word "in," which has a weak semantic relation with the related word "contains." Such trigger words are difficult to find. We found that relations related to the location are prone to weakly associated trigger words. Thus, we have summarized these patterns corresponding to 7 specific relations, as shown in Table 1.

3.3. Relation Extractor. The single sentence contains less semantic information in distantly supervised RE, and most RE methods extract global context features of sentences, while ignoring the implicit information between the entity pair and the different words in the sentence. Thus, we introduce the interaction representation between entity pairs and sentences as a supplementary feature for the RE task. We use BiLSTM to obtain the semantic representation of the sentence and mentioned entity pair, respectively. Then, the

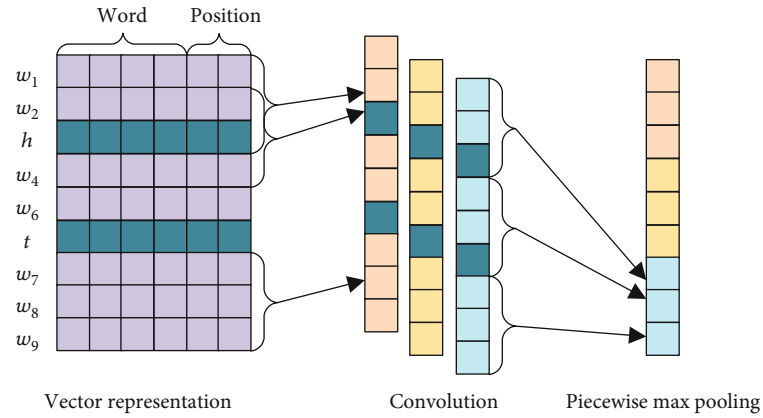

Figure 4: The architecture of the PCNNs module.

interaction representation between entity pairs and sentences is calculated. Afterward, we apply PCNN to obtain sentence features and concatenate these features and interaction representation as the entire sentence representation. These entire sentence representations are assigned different weights by sentence-level attention to obtain bag representation. Finally, the bag representation is fed into the softmax classifier to predict the relation.

3.3.1. Sentence Feature with PCNN. As a sentence encoder, PCNN has performed satisfactorily in the RE tasks and captures structural information between two entities [8, 10]. Given two entities (h, t), the PCNNs divide the sentence into three segments based on the location of the entity pair: the piece between the head entity h and the tail entity t, the piece before h, and the piece after t. These three parts relate to characters inside or around two entities, respectively, and are treated as one internal context and two external contexts. Figure 4 displays the architecture of the PCNNs module. We concatenate word embedding and position embedding as input to PCNN.

(1) Position Embedding. Zeng et al. [42] first propose position embedding to specify entity pairs. Position embedding is defined as the combination of relative distances from the current word to the head entity h and tail entity t. For example, the relative distances of "founded" in sentence S1 to head entity "Microsoft" and tail entity "Bill Gates" are 4 and -1, respectively. The initial embedding matrix is randomly generated. Then, we look up the vector of two relative distances in the embedding matrix. Position embedding of the word relative to the entity pair h and t is denoted as $\mathrm{PE}_{i,1}$ and $\mathrm{PE}_{i,2}$, respectively.

We concatenate word embedding WE_i and two position embedding $\mathrm{PE}_{i,1}$ and $\mathrm{PE}_{i,2}$ to form a word representation, i.e., $w_i = [\mathrm{WE}_i ; \mathrm{PE}_{i,1} ; \mathrm{PE}_{i,2}]$, where $[\mathrm{WE}_i ; \mathrm{PE}_{i,1} ; \mathrm{PE}_{i,2}]$ represents the vertical connection of vector WE_i, $\mathrm{PE}_{i,1}$, and $\mathrm{PE}_{i,2}$. Then, the sentence representation will be $s = [w_1 ; w_2 ; \cdots \cdots ; w_m]$, where $s \in \mathbb{R}^{m \times d}$, $d = d_w + d_p * 2$ denotes the dimension of the final word vector, and d_p represents the dimension of the position vector.

As shown in Figure 4, the PCNN is mainly composed of two parts. One is the convolutional layer, which uses convolution operations to flexibly extract the local features of the sentence. The calculated representation of the i-th filter of the convolutional layer is $M_i = \mathrm{CNN}_i(s)$, where $i = 1, 2, \cdots \cdots, d^c$. Another is piecewise max pooling. The output of the convolutional filter M_i is divided into three segments according to the position of the entity pair, and these three pieces are denoted as $M_{i,1}$, $M_{i,2}$, and $M_{i,3}$. Then, the piecewise max-pooling finds the maximum value of each segment separately, which is defined as

$$z_i = \max\left(M_{i,j}\right), 1 \le i \le d^c, 1 \le j \le 3. \tag{8}$$

After piecewise max pooling, we can obtain a 3-dimensional vector $z_i = [z_{i,1}, z_{i,2}, z_{i,3}]$. Then, we concatenate all the vectors to represent as $z_{1:d^c}$ and use a nonlinear function, such as hyperbolic tangent. Finally, the output of PCNNs is the sentence feature, which is expressed as

$$Q_{pcnn} = \tanh\left(z_{1:d^c}\right). \tag{9}$$

3.3.2. Interaction Representation. To get better sentence encoding, Jat et al. [33] used word-level attention related to entity pairs and relations to learn attention weights, and the quality of the attention weights obtained is closely related to the relation vectors. For obtaining more semantic information in RE tasks, we propose the interaction representation using the word-level attention mechanism-based entity pairs. The difference from the method proposed by Jat et al. [33] is that we do not use the relation vectors when we learn attention weights. Our approach is more like machine reading comprehension [43]. Given entity pairs and sentences, the entity pair are regarded as crucial words of the query. We use the query to find the answer in the sentence, find the words related to the entity pair, and get the matching score matrix. Then, the attention weights and sentence representation are calculated. Specifically, as shown in Figure 5, we use BiLSTM to obtain the vector representation of the entity pairs and the sentences, respectively. Then, the interaction representation between entity pairs and sentences is calculated. As we all know, LSTMs are suitable for processing

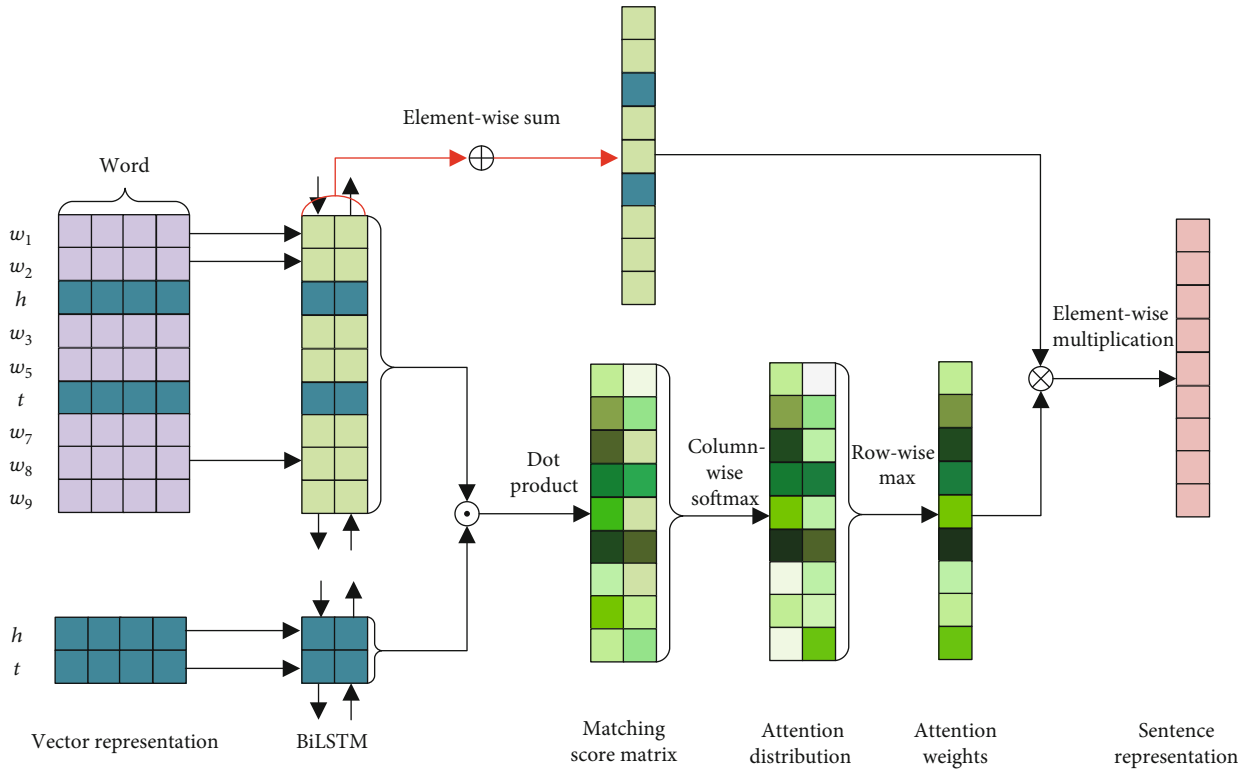

FIGURE 5: The architecture of the interaction representation.

sequential data such as text and speech because they retain a hidden state vector that changes with the input data at each step. Given the word sequences $\{a_1, a_2, \cdots \cdots, a_m\}$, the vector sequence is denoted as $\{\mathrm{WE}_1, \mathrm{WE}_2, \cdots \cdots, \mathrm{WE}_m\}$. We use BiLSTM to extract the semantic features, where the encoding representation of word sequences consists of vertically connecting between forward and backward hidden states. At the time step t, the hidden layer states of the forward LSTM and the backward LSTM can be briefly described as $\overrightarrow{h_t} = \overrightarrow{\mathrm{LSTM}}(\mathrm{WE}_t)$ and $\overleftarrow{h_t} = \overleftarrow{\mathrm{LSTM}}(\mathrm{WE}_t)$, respectively. Then, we concatenate the forward hidden state and the backward hidden state as follows:

$$h_t = \left[\overrightarrow{h_t}; \overleftarrow{h_t}\right]. \tag{10}$$

Our model uses two BiLSTMs to extract the context representation and semantic features of word sequences. The vectors for entity pairs and sentences obtained by BiLSTMs are expressed as h_{en} and h_{sen}, respectively. After getting the contextual embeddings of the entity pair and sentence, we calculate the match score for each word between the entity pair and the sentence, which is represented as follows:

$$H(i, j) = h_{en}(i) \cdot h_{sen}(j), \tag{11}$$

where matrix $H \in \mathbb{R}^{m \times m_{en}}$ represents the matching score of the context representation between the entity pair and the sentence. m_{en} denotes the number of entities in an entity pair. m is the length of the sentence. The matrix $H(i, j)$ represents the sum of the dot product of the context representation between the i-th word in the entity pair and the j-th word in the sentence.

After obtaining the matching score matrix H, we use a column-wise softmax function to obtain probability distributions in each column, which calculates the importance of each word in the entity pair to each word in the sentence. Thus, the calculation process of the word-level attention distribution based on entity pairs is as follows:

$$\alpha(t) = \mathrm{softmax}(H(1, t), \cdots, H(m, t)), \tag{12}$$

$$\alpha = [\alpha(1), \alpha(2), \cdots, \alpha(m_{en})], \tag{13}$$

where $\alpha(t)$ indicates the probability distributions obtained by using the softmax function for all values in the column t.

We select the maximum value of each row from the probability distributions α and perform the softmax function on the result to obtain the final attention weights, which calculates the importance of the entity pair to each word in the sentence. The final attention weights q are computed as:

$$q = \mathrm{softmax}\left(\max_{t=1 \cdots m} \alpha(t)\right). \tag{14}$$

The interaction representation is expressed as:

$$Q_{IR} = q \otimes \tilde{h}_{sen}, \tag{15}$$

$$\tilde{h}_{sen} = \overrightarrow{h_s} \oplus \overleftarrow{h_s}, \tag{16}$$

where $\oplus$ represents the element-wise sum. Here, we use the element-wise sum to combine the forward and backward hidden layer states of the word sequence in the sentence, which is expressed as $\tilde{h}_{sen}$. $\otimes$ represents the element-wise multiplication.

As shown in Figure 2, we concatenate the PCNN-based sentence feature Q_{pcnn} with the interaction representation Q_{IR} to get the final sentence representation Q.

$$Q = \left[Q_{pcnn}; Q_{IR}\right]. \tag{17}$$

3.3.3. Sentence-Level Attention. Similar to the previous study [10], the sentence-level attention model is used to obtain the bag representation for RE. The bag matrix G consisting of sentence representations is described as:

$$G = [Q_1; Q_2; \cdots \cdots; Q_l]. \tag{18}$$

The attention weights of the sentences in the bag are calculated as:

$$u_i = Q_i A r, \tag{19}$$

$$\beta_i = \frac{\exp(u_i)}{\sum_j \exp(u_j)}, \tag{20}$$

where u_i is the correlation degree between sentences and relation labels. A is a weighted diagonal matrix. r is the query vector associated with relations which indicates the representation of relations. β_i is the attention weight of the i-th sentence in the bag.

The final bag embedding $\widehat{G}$ is computed as a weighted sum of these sentence representation:

$$\widehat{G} = \sum_i \beta_i Q_i. \tag{21}$$

3.3.4. Objection Function and Optimization. Given an entity pair and all the sentences mentioning these two entities $(h, t, \{s_1, s_2, \cdots \cdots s_c\})$,$B = \{s_1, s_2, \cdots \cdots, s_n\}$ represents a bag, which is a collection of sentences with the same pair of entities. We define the conditional probability $p(r \mid B, \theta)$ through the softmax function to calculate the confidence of each possible relation:

$$p(r \mid B, \theta) = \frac{\exp(o_r)}{\sum_{k=1}^{n_r} \exp(o_k)}, \tag{22}$$

where n_r denotes the number of relations. o represents the final output of the neural network, which is defined as follows:

$$o = M\widehat{G} + d, \tag{23}$$

where M is the transformation matrix. d is a bias vector.

Finally, we define the objection function using cross-entropy.

$$J(\theta) = \sum_{i=1}^{n} \log p(r_i \mid B_i, \theta), \tag{24}$$

where B_i is the i-th bag in the training data, r_i is a possible relation label in the bag B_i. θ represents all parameters of our model. To solve the optimization problem, similar to previous studies [8, 10], we apply stochastic gradient descent (SGD) to minimize the objection function.

4. Experiments

4.1. Dataset and Evaluation Metrics. We evaluate our method on a widely used dataset (available at http://iesl.cs.umass.edu/riedel/ecml/), which is generated by Riedel et al. [5]. This dataset is developed by aligning the triples consisting of entity pairs and relations in Freebase with the New York Times NYT corpus. The dataset includes 53 relations containing the label "NA," which indicates there is no relation between two entities. The statistics of the used dataset are shown in Table 2.

Similar to previous work [8, 10], we use the held-out evaluation to evaluate our model, which evaluates our model by comparing the predictions in the testing set with the relational facts in Freebase. In our experiments, we use precision/recall curves, the highest F1 value, and P@N metrics to evaluate the model in all aspects.

4.2. Parameter Settings

4.2.1. Word and Entity Embedding. Similar to previous works [10, 38], we apply the word2vec (https://code.google.com/p/word2vec/) tool to train the word embeddings and entity embeddings in the NYT corpus. To better complete the RE task, we use word embeddings obtained by word2vec as the initial representation of the word and add position embedding according to the relative distance of the word to the two entities in the sentence.

In our experiments, we tune our model adopting threefold validation. The whole parameter settings of our model are listed in Table 3. For the parameters of our model, we set the dimension of word embedding and entity embedding $d_w = 50$, the position embedding $d_p = 5$, the number of feature maps $d^h = 230$, the window size $l = 3$, the learning rate $\lambda = 0.01$, the dropout probability $p = 0.5$, and the similarity threshold $\delta = 0.6$. The batch size is fixed to 160.

4.3. Baselines. To evaluate the effect of our model, we compared the proposed model with five strong baselines (traditional feature-based methods: Mintz, MultiR, and MIML; Neural Network Approaches: PCNN+ATT, PCNN+RL).

Traditional feature-based methods

(i) *Mintz* is a traditional feature-based RE method proposed by Mintz et al. [3]

(ii) *MultiR* is a novel approach presented by Hoffmann et al. [6] for multi-instance learning, which is to

Table 2: Statistics of the datasets.

Dataset	Sentences	Entity pairs	Relation facts
Training set	522,611	281,270	18,252
Testing set	172,448	96,678	1,950

Table 3: Parameter settings.

Parameter name	Parameter setting
Word dimension	50
Position dimension	5
Number of feature maps	230
Window size	3
Learning rate	0.01
Dropout probability	0.5
Similarity threshold	0.6
Batch size	160

resolve overlapping relations on the distantly supervised RE

(iii) *MIML* [7] is an approach to multi-instance multilabel learning for RE, which employs a graphical model with latent variables

Neural network approaches

(i) *PCNN + ATT* [10] is currently the most advanced neural network approaches in distantly supervised RE, which applies the PCNN module to obtain sentence feature, and sentence-level attention to alleviate the weights of those noisy instances

(ii) *PCNN + RL* [38] use RL to redistribute wrong-labeled instances into the negative set and PCNN+ATT module to generate bag encoding

In order to fully demonstrate the effectiveness of our proposed methods, we have designed three different methods for the PCNN+ATT model.

(i) *PCNN + ATT + SS* employs the pattern method based on relation trigger word to filter the wrong labels, and the filtered data is feed to the PCNN + ATT extractor for training and testing

(ii) *PCNN + ATT + IR* uses the interaction representation between entity pairs and sentences as a supplementary feature for the RE

(iii) *PCNN + ATT + SS + IR* combines the above two methods

4.4. Experimental Results and Analysis

4.4.1. Results Analysis and Evaluation

(1) Similarity Threshold Selection (Performance Comparison of Different Semantic Similarity Thresholds). To select the right value for the semantic similarity threshold, we compare the precision of the proposed method under different similarity thresholds. Figure 6 shows the precisions of our method when the value of the semantic similarity threshold varies from 0 to 0.9. From Figure 6, we can observe the following.

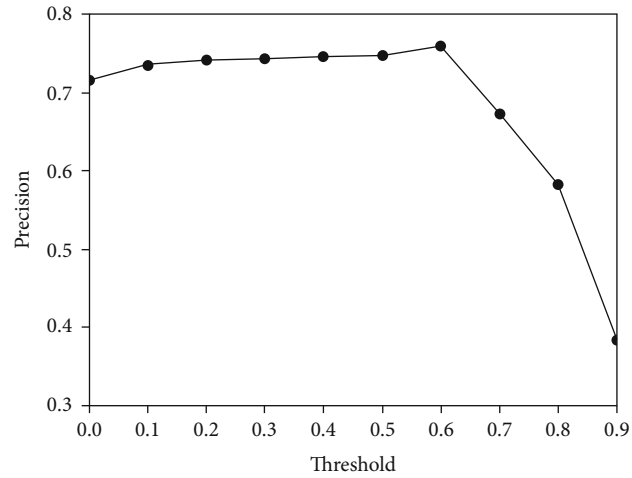

Figure 6: The precision comparison with different similarity thresholds.

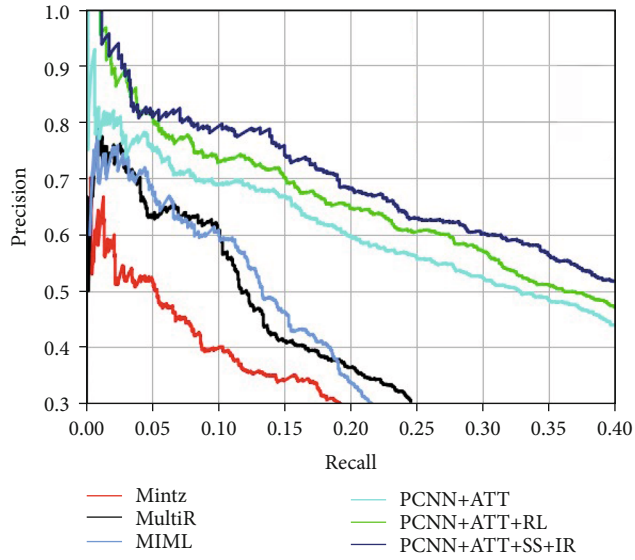

Figure 7: The precision/recall curves of our model and the strong baselines.

In Figure 6, our method achieves the highest precision when the semantic similarity threshold is 0.6. When the threshold is less than 0.6, the dataset obtained by our method may contain more noisy sentences, which affects the performance of RE. When the threshold is gradually increased, more and more noisy data is removed, and the precision of RE becomes higher and higher. However, when the threshold is greater than 0.6, some correct instances are filtered out, which causes the dataset to become smaller and contains less useful information, and it is impossible to train the relation extractor that performs well enough. When the threshold is

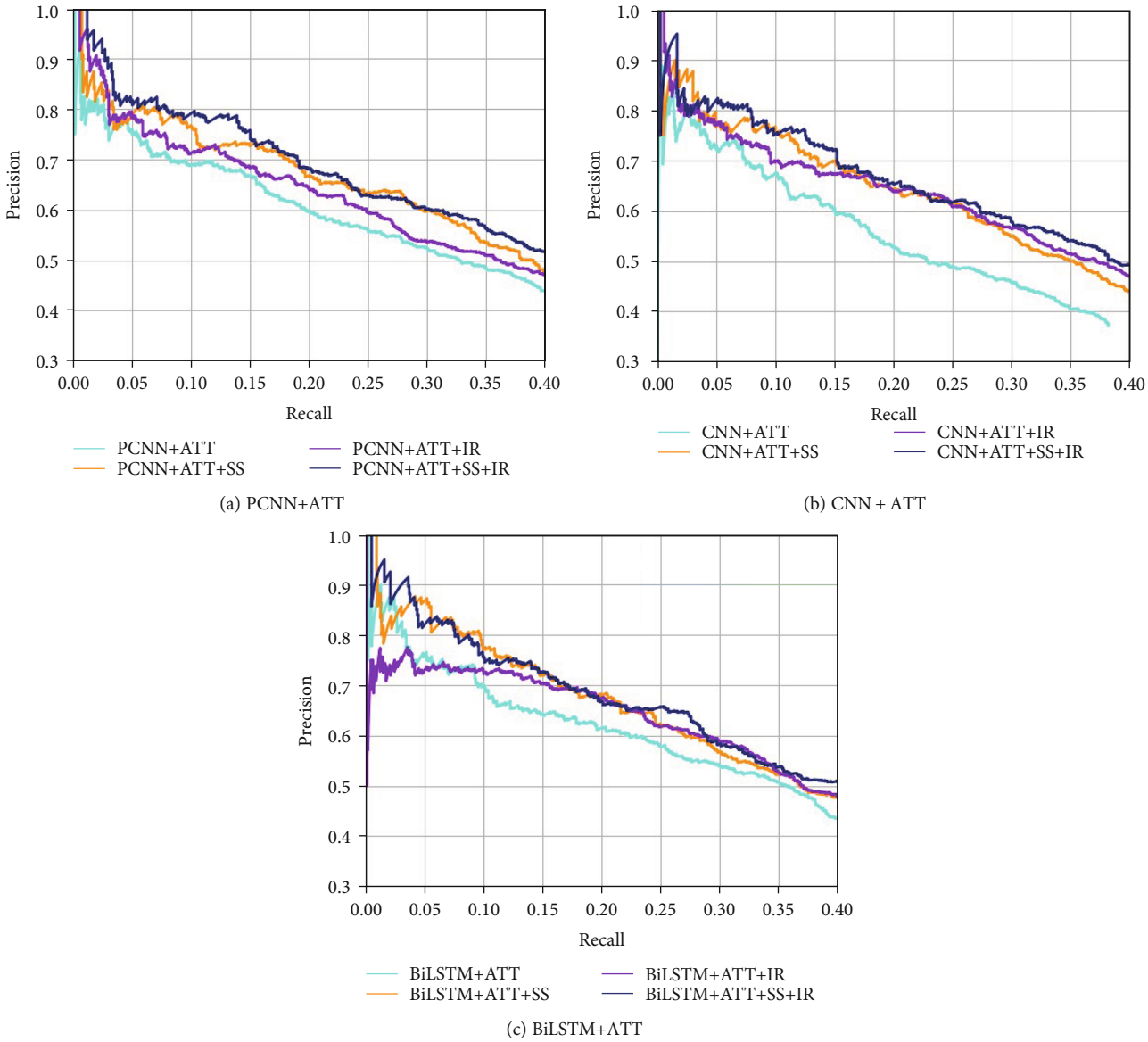

Figure 8: Performance comparison based on different sentence encoders.

greater than 0.6, the precision decreases with the increase of the threshold value. Thus, we set the semantic similarity threshold to 0.6 in the paper.

(2) Precision/Recall Curves (Performance Comparison of Different Methods). Figure 7 shows the precision/recall curves of our method and five strong baselines, we have the following observations.

In Figure 7, we compare three traditional feature-based methods, including Mintz, MultiR, and MIML, to our proposed PCNN+ATT + SS + IR via precision/recall curves. It can be seen that the precision of PCNN+ATT + SS + IR is always better than the three traditional feature-based methods in the same recall. At the same time, with the same precision, the PCNN+ATT + SS + IR achieves the best recall in these several methods. Moreover, comparing the other two neural network approaches—PCNN+ATT and PCNN+ATT + RL with three traditional feature-based methods, it is clear that neural network approaches are much better than traditional feature-based methods. It is worth noting that the three traditional feature-based methods use the NLP tool to extract features. However, our method and the other two neural network approaches use the neural network to automatically extract features. The results show that our method and neural network approaches can effectively solve the error propagation and accumulation problems in NLP tools and improve the performance of distantly supervised relation extractors. We compare the other two neural network approaches with the PCNN+ATT + SS + IR via precision/recall curves. In most areas of the curve, PCNN+ATT + SS + IR outperforms PCNN+ATT and PCNN+ATT + RL according to precision and recall. Compared with the five-strong baseline methods, the results

Table 4: F1 and P@N for our proposed methods and baselines.

P@N (%)	100	200	300	Mean	F1 (%)
Mintz	51.82	50.00	44.84	48.88	24.29
MultiR	70.30	65.17	61.79	65.76	27.48
MIML	70.90	62.86	60.97	64.91	25.31
PCNN+ATT	76.24	71.14	69.10	72.16	42.57
PCNN+ATT + RL	81.19	77.11	73.75	77.35	44.29
PCNN+ATT + SS	80.20	72.14	70.43	74.26	44.85
PCNN+ATT + IR	78.22	75.12	72.09	75.15	44.01
PCNN+ATT + SS + IR	82.19	78.61	72.09	77.63	46.90
CNN + ATT	76.24	70.65	66.45	71.11	38.29
CNN + ATT + SS	76.24	71.64	66.78	71.55	42.34
CNN + ATT + IR	77.23	73.63	68.77	73.21	43.48
CNN + ATT + SS + IR	80.20	74.63	67.77	74.20	46.39
BiLSTM+ATT	75.25	69.65	64.45	69.78	42.21
BiLSTM +ATT + SS	83.16	73.63	69.43	75.41	43.94
BiLSTM +ATT + IR	76.24	73.13	73.09	74.15	44.13
BiLSTM +ATT + SS + IR	82.18	75.12	69.43	75.58	45.23

indicate that our method can effectively reduce the sentences with the wrong label and extract more significant features to improve the performance of the RE.

(3) Precision/Recall Curves (Performance Comparison of Our Methods under Different Sentence Encoders). To verify the effectiveness and robustness of our proposed model in different sentence encoders, we replace the PCNN+ATT module with CNN + ATT and BiLSTM+ATT, respectively. Figure 8 shows the precision-recall curves of neural network methods with different sentence encoders (PCNN+ATT, CNN + ATT, and BiLSTM+ATT), which indicates the following:

(1) We can observe that using the sentence selection and interaction representation can boost the performance of PCNN/CNN/BiLSTM+ATT, especially our models PCNN/CNN/BiLSTM+SS + IR achieve the highest precision in the corresponding sentence encoders over the entire range of recall. Our proposed hierarchical sentence selector can effectively reduce noise sentences. Thus, the sentence selection applying the pattern based on the relation trigger is effective for filtering the noisy sentences, and the interaction representation between entity pairs and sentences can provide useful semantic information for relation prediction. Besides, the results demonstrate the effectiveness and robustness of the proposed models in different sentence encoders

(2) Since our proposed sentence selection (PCNN/CNN/BiLSTM+ATT + SS) and interaction representation (PCNN/CNN/BiLSTM+ATT + IR) is used to solve two different problems in distantly supervised RE, so it is not appropriate to directly compare the experimental results of these two methods. PCNN/CNN/BiLSTM+ATT + SS performs better than PCNN/CNN/BiLSTM+ATT. Remarkably, when the recall is the same, PCNN/CNN/BiLSTM+ATT + SS + IR achieves much higher precision than PCNN/CNN/BiLSTM+ATT + IR. These results indicate that RE tasks have the wrong labeling problem and sentence selector using the pattern method based on the relation trigger can effectively remove the noisy sentences

(3) In the same recall, PCNN/CNN/BiLSTM+ATT + IR has higher precision than PCNN/CNN/BiLSTM+ATT. In addition, PCNN/CNN/BiLSTM+ATT + SS + IR outperforms PCNN/CNN/BiLSTM+SS, respectively. Therefore, we conclude that the proposed interaction representation can provide additional semantic information and generate better sentence embedding to improve the performance of the RE

(4) The F1 Value. In Table 4, the methods based on sentence selection (PCNN/CNN/BiLSTM+ATT + SS) and the methods based on interaction representation (PCNN/CNN/BiLSTM+ATT + IR) obtain a higher F1 value than the original sentence encoder (PCNN/CNN/BiLSTM+ATT), respectively. The results prove that these two methods can effectively improve the performance of RE. CNN + ATT + SS + IR obtains the highest F1 value in sentence encoders based on CNN + ATT, and the F1 value of CNN + ATT + SS + IR is 8.1 percentage points higher than that of the original sentence encoder CNN + ATT. Similarly, BiLSTM+ATT + SS + IR also achieves the highest F1 value, which is 3.0% higher than BiLSTM+ATT, 1.3% higher than BiLSTM+ATT + SS, and 1.1% higher than BiLSTM+ATT+ IR. In addition, proposed PCNN+ATT + SS + IR obtains the highest F1 value in all neural baselines, which are 4.3% and 2.9% higher than PCNN+ATT and PCNN+ATT + RL, respectively. These results demonstrate that the proposed method can remove the sentences with the wrong label and provide useful sentence features.

(5) P@N Metrics. Following Lin et al. [10], we employ the P @ N metric to evaluate strong baselines and our proposed methods as shown in Table 4. We report the P@100, P@300, P@500, and their mean value. It can be found that the proposed method can improve the results of the original sentence encoder PCNN/CNN/BiLSTM+ATT to some extent and significantly better than the traditional feature-based methods. Moreover, PCNN+ATT + SS + IR achieves the highest precision, which is 6.0, 7.5, 3.0, and 5.5 points higher than PCNN+ATT in P@100, P@200, P@300, and mean value, respectively. Based on these results, we conclude that our model can effectively select higher quality sentences and make full use of implicit information to provide better sentence embedding for RE tasks.

4.4.2. Case Study and Discussion. Table 5 shows some examples of semantic similarity between relation phrases and relation trigger words in the testing data. The first column is a

Table 5: Some examples of semantic similarity in the NYT corpus.

Relation	Sentence	Words with cosine similarity
/Business/company/founders	That issue inspired a young subscriber named **Bill Gates**, who later founded **Microsoft**, to begin programming for the computer.	Founded (0.655194)
/Location/country/capital	I eventually fly from Nairobi to Dubai to **Asmara**, the capital of **Eritrea**.	Capital (1.0)
/People/person/religion	**Fazlur Rahman**, a brilliant and deeply religious Pakistani scholar of **Islam**, had to flee his native land for the University of Chicago.	Religious (0.832861)
/Location/neighborhood/neighborhood_of	Last year, pacific retirement services, a nonprofit organization based in Medford, ore., began construction on the Mirabella, a continuing-care community in the South Lake **Union** neighborhood of **Seattle**.	Neighborhood (1.0)
/Business/company/founders	**Bill Gates**, **Microsoft**'s chairman, contend that the bullish case for Microsoft stock is compelling	Chairman (0.597136)
/Location/location/contains	Campbell was Williams's hometown friend from their childhood in **Wellington**, **New Zealand.**	From (0.277430)

relation which is the label annotated for the sentence by DS. The second column is the sentence containing the entity pair. And we highlight the entity pairs with bold formatting. The last column is the word and score with the highest semantic similarity in the corresponding sentence.

From Table 5, we find that words with higher semantic similarity are usually closely related to the relation between entity pairs. For example, the semantic similarity score for the word "founded" and the relation /business/company/-founders is higher than other words in the first sentence in Table 5. Similarly, the words "capital," "religious," and "neighborhood" have the highest semantic similarity scores in their corresponding sentences, respectively. The results show that our method can extract the trigger words related to the related phrases and filter the wrong label sentences.

In addition, we have examined some sentences filtered by the pattern method based on the relation trigger words. The sentences listed in the last two rows of Table 5 are typical examples: the semantic similarity score between the word "chairman" in the first sentence and the relation /business/-company/founders is 0.597136, whereas we set the semantic similarity threshold to 0.6, which can filter this wrong label sentence. The relation label of the last sentence containing the entity pair (New Zealand, Wellington) is correct. But, the highest score in the last sentence is 0.277430. According to the pattern method based on the relation trigger words, we wrongly filter the correct label sentences. It is common sense for us that "in Wellington, New Zealand" means "New Zealand contains Wellington." However, the plain text implicitly expresses the relation /location/location/contains and does not provide a word that is closely related to the relation phrase. Such trigger words are difficult to find. Similar examples are shown in Table 1. For example, in Table 1, it is easy for us to find that the sentence "..., said Mr.Cho, 25, who wanted to Seoul, South Korea, and educated at a boarding school in Scotland." in the second line and the sentence "She said the state's division of special revenue was investigating the incident, which took place at the studios of Wtic television in Hartford, where the drawing is televised." in the third line express relation /location/location/contains and /broadcast/content/location, respectively. However, it is difficult for the sentence selector to find trigger words related to these relations. Thus, it is necessary to select the patterns corresponding to the specific relations, and it also proves that the pattern method can effectively remove the noisy sentences.

5. Conclusions and Future Work

In this paper, we propose a novel relation extraction (RE) method based on sentence selection and interaction representation. The model has two modules: sentence selector and relation extractor. The sentence selector applying the pattern method based on the relation trigger words can remove more noisy sentences and select higher quality sentences, which alleviates the wrong labeling problem in distant supervision (DS). The relation extractor uses the interaction representation between entity pairs and sentences as a supplementary feature for RE to make full use of the implicit information existing in the sentence. The experimental results indicate that the proposed method outperforms previous state-of-the-art baselines and can effectively improve the performance of RE.

In our future work, we intend to explore the following research: (1) our method only considers the relation of a single sentence containing two entities. However, the sentence may contain multiple entities. There may be implicit associations between these entities, which can be used to improve the performance of RE. (2) Most existing RE merely focuses on predicting relation from monolingual data, ignoring the rich information in multilingual corpus. We hope to apply the proposed model to multilingual data.

Data Availability

Previously reported [DATA TYPE] data were used to support this study and are available at [S. Riedel, L. Yao, and A. McCallum, "Modeling relations and their mentions without labeled text," in Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 148–163, 2010.]. These prior studies (and datasets) are cited at relevant places within the text as references [5].

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant (No. 61772152), National Key R&D Program of China (No. 2018YFC0806800), the Technical Basic Research Project (No. JSQB2017206C002 and No. JCKY2019604C004), Pre-research Project (No. 10201050201), the Project funded by China Postdoctoral Science Foundation under Grant (No. 2019M651262), the Youth Fund Project of Humanities and Social Sciences Research of the Ministry of Education of China under Grant (No. 20YJCZH172), and the Postdoctoral Foundation of Heilongjiang Province under Grant (No. LBH-Z19015).

References

[1] G. Weikum and M. Theobald, "From information to knowledge: harvesting entities and relationships from web sources," *Proceedings of the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pp. 65–76, 2010.

[2] H. Sun, H. Ma, W. T. Yih, C. T. Tsai, J. Liu, and M. W. Chang, "Open domain question answering via semantic enrichment," in *Proceedings of the 24th International Conference on World Wide Web*, pp. 1045–1055, Florence, Italy, 2015.

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

[4] C. Ru, J. Tang, S. Li, S. Xie, and T. Wang, "Using semantic similarity to reduce wrong labels in distant supervision for relation extraction," *Information Processing & Management*, vol. 54, no. 4, pp. 593–608, 2018.

[5] S. Riedel, L. Yao, and A. McCallum, "Modeling relations and their mentions without labeled text," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 148–163, Berlin, Heidelberg, 2010.

[6] R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, and D. S. Weld, "Knowledge-based weak supervision for information extraction of overlapping relation," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pp. 541–550, Portland, Oregon, USA, 2011.

[7] 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*, pp. 455–465, Jeju Island, Korea, 2012.

[8] 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*, pp. 1753–1762, Lisbon, Portugal, 2015.

[9] H. Wen, X. Zhu, L. Zhang, and F. Li, "A gated piecewise CNN with entity-aware enhancement for distantly supervised relation extraction," *Information Processing & Management*, vol. 57, no. 6, article 102373, 2020.

[10] Y. Lin, S. Shen, Z. Liu, H. Luan, and M. Sun, "Neural relation extraction with selective attention over instances," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pp. 2124–2133, Berlin, Germany, 2016.

[11] X. Wang, X. Han, Y. Lin, Z. Liu, and M. Sun, "Adversarial multi-lingual neural relation extraction," in *Proceedings of the 27th International Conference on Computational Linguistic*, pp. 1156–1166, Santa Fe, New Mexico, USA, 2018.

[12] M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni, "Open information extraction from the web," in *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, vol. 7, pp. 2670–2676, Yokohama, Japan, 2007.

[13] S. Zhao and R. Grishman, "Extracting relations with integrated information using kernel methods," in *Proceedings of the 43rd annual meeting of the association for computational linguistics*, pp. 419–426, Ann Arbor, Michigan, 2005.

[14] N. Kambhatla, "Combining Lexical, Syntactic, and Semantic Features with Maximum Entropy Models for Extracting Relations," in *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, p. 22, Barcelona, Spain, 2004.

[15] 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*, pp. 1768–1777, Copenhagen, Denmark, 2017.

[16] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.

[17] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41, no. 6, pp. 391–407, 1990.

[18] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, p. 788, 1999.

[19] V. T. Phi, J. Santoso, M. Shimbo, and Y. Matsumoto, "Ranking-based automatic seed selection and noise reduction for weakly supervised relation extraction," in *Proceedings 56th Annual Meeting of the Association for Computational Linguistics*, pp. 89–95, Melbourne, Australia, 2018.

[20] Y. He, Z. Li, G. Liu et al., "Bootstrapped multi-level distant supervision for relation extraction," in *International Conference on Web Information Systems Engineering*, pp. 408–423, Dubai, United Arab Emirates, 2018.

[21] J. Qu, D. Ouyang, W. Hua, Y. Ye, and X. Li, "Distant supervision for neural relation extraction integrated with word attention and property features," *Neural Networks*, vol. 100, pp. 59–69, 2018.

[22] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," *Advances in neural information processing systems*, pp. 2787–2795, 2013.

[23] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Twenty-ninth AAAI conference on artificial intelligence*, pp. 2181–2187, Austin, Texas, USA, 2015.

[24] P. Zhou, J. Xu, Z. Qi, H. Bao, Z. Chen, and B. Xu, "Distant supervision for relation extraction with hierarchical selective attention," *Neural Networks*, vol. 108, pp. 240–247, 2018.

[25] H. Zhao, R. Li, X. Li, and H. Tan, "CFSRE: context-aware based on frame-semantics for distantly supervised relation extraction," *Knowledge-Based Systems*, vol. 210, p. 106480, 2020.

[26] T. Sun, C. Zhang, Y. Ji, and Z. Hu, "MSnet: multi-head self-attention network for distantly supervised relation extraction," *IEEE Access*, vol. 7, pp. 54472–54482, 2019.

[27] 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*, pp. 1257–1266, Brussels, Belgium, 2018.

[28] H. Ye and Z. Luo, "Deep ranking based cost-sensitive multi-label learning for distant supervision relation extraction," *Information Processing & Management*, vol. 57, no. 6, article 102096, 2020.

[29] S. Mitra, S. Saha, and M. Hasanuzzaman, "A multi-view deep neural network model for chemical-disease relation extraction from imbalanced datasets," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 11, pp. 3315–3325, 2020.

[30] Y. Shi, Y. Xiao, P. Quan, M. Lei, and L. Niu, "Distant supervision relation extraction via adaptive dependency-path and additional knowledge graph supervision," *Neural Networks*, vol. 134, pp. 42–53, 2020.

[31] X. Ouyang, S. Chen, and R. Wang, "Semantic enhanced distantly supervised relation extraction via graph attention network," *Information*, vol. 11, no. 11, p. 528, 2020.

[32] V. T. Phi, J. Santoso, V. H. Tran, H. Shindo, M. Shimbo, and Y. Matsumoto, "Distant supervision for relation extraction via piecewise attention and bag-level contextual inference," *IEEE Access*, vol. 7, pp. 103570–103582, 2019.

[33] S. Jat, S. Khandelwal, and P. Talukdar, *Improving Distantly Supervised Relation Extraction using Word and Entity Based Attention*, 2018.

[34] Z. Geng, G. Chen, Y. Han, G. Lu, and F. Li, "Semantic relation extraction using sequential and tree-structured LSTM with attention," *Information Sciences*, vol. 509, pp. 183–192, 2020.

[35] Y. Ye, Y. Feng, B. Luo, Y. Lai, and D. Zhao, "Integrating relation constraints with neural relation extractors," 2019, https://arxiv.org/abs/1911.11493.

[36] J. Feng, M. Huang, L. Zhao, Y. Yang, and X. Zhu, "Reinforcement learning for relation classification from noisy data," in *Thirty-Second AAAI Conference on Artificial Intelligence*, New Orleans, Louisiana, USA, 2018.

[37] X. Zeng, S. He, K. Liu, and J. Zhao, "Large scaled relation extraction with reinforcement learning," in *Thirty-Second AAAI Conference on Artificial Intelligence*, vol. 2, p. 3, New Orleans, Louisiana, USA, 2018.

[38] P. Qin, W. Xu, and W. Y. Wang, "Robust distant supervision relation extraction via deep reinforcement learning," in *Proceedings 56th Annual Meeting of the Association for Computational Linguistics*, pp. 2137–2147, Melbourne, Australia, 2018.

[39] T. Sun, C. Zhang, Y. Ji, and Z. Hu, "Reinforcement learning for distantly supervised relation extraction," *IEEE Access*, vol. 7, pp. 98023–98033, 2019.

[40] T. Chen, N. Wang, M. He, and L. Sun, "Reducing wrong labels for distantly supervised relation extraction with reinforcement learning," *IEEE Access*, vol. 8, pp. 81320–81330, 2020.

[41] G. Wang, W. Zhang, R. Wang et al., "Label-free distant supervision for relation extraction via knowledge graph embedding," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2246–2255, Brussels, Belgium, 2018.

[42] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, "Relation classification via convolutional deep neural network," in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics*, pp. 2335–2344, Dublin, Ireland, 2014.

[43] M. Hu, F. Wei, Y. Peng, Z. Huang, N. Yang, and D. Li, "Read + verify: machine reading comprehension with unanswerable questions," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 6529–6537, 2019.