IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 25, 2019, accepted April 21, 2019, date of publication April 25, 2019, date of current version May 6, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2913316

MSnet: Multi-Head Self-Attention Network for Distantly Supervised Relation Extraction

TINGTING SUN, CHUNHONG ZHANG, YANG JI, AND ZHENG HU
State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

Corresponding author: Chunhong Zhang (zhangch@bupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61421061, Grant 61602048, Grant 61601046, and Grant 61520106007, in part by the Project of Hainan Passenger Behavior Intelligence Analysis Platform and Precise Service Mining Prediction under Grant ZDKJ201808, and in part by the Beijing University of Posts and Telecommunications, School of Information and Communication Engineering, Excellent Graduate Students Innovation Fund in 2016.

ABSTRACT Distant supervision for relation extraction is a task of recognizing semantic relations between entities in a large amount of plain text weakly supervised by external knowledge bases, which can benefit many NLP applications, such as knowledge graph completion and question answering. While it significantly alleviates the expensive cost for data labeling, it severely suffers from noisy labels. In this paper, we propose a Multi-head Self-attention Network (MSNet)-based label denoising method for relation extraction. More specifically, we encode the words, entities and their positions information into contextual embeddings via a multi-head self-attention mechanism, then extract the discriminative sentence features with max pooling operation. MSNet can capture the inherent structure of a sentence and model the relatedness between two words without regard to their distance. Moreover, we adopt a novel label confidence learning method to correct the noisy labels. A latent label is predicted step by step during training as the ground-truth according to a curriculum function of label confidence. This label denoising mechanism gradually incorporates the obtained latent label of easy relation patterns into later latent label prediction of hard patterns, which makes latent label consistent learning more reliable. To verify the effectiveness of our proposed method, in addition to the widely used PCNN-based architecture, we also perform the experiment on BiLSTM model as a comparison. The results demonstrate that our approach can outperform the state-of-the-art systems on the popular evaluation dataset.

INDEX TERMS Relation extraction, distant supervision, multi-head self-attention, label denoising.

I. INTRODUCTION

Relation Extraction (RE) is defined as a task of generating relation triple facts from plain texts, which is widely used to facilitate a lot of Natural Language Processing (NLP) tasks including knowledge base construction and question answering. As the fully supervised RE approaches are limited by the consuming and labour intensive labeled training set, *Distant supervision* strategy [1] is proposed as a promising approach to automatically create training data via aligning Knowledge Bases (KBs) with texts. The basic assumption of distant supervision is that if two entities e_1 and e_2 have a relation r in KBs, then all the sentences in corpus that contain these two entities will express this specific relation and will be labeled as the training instances of r.

However, this strong assumption always makes the training data suffer from wrong labeling problem due to the in-completion or inherent multi-relation nature of the exploited KBs. For example, as shown in Fig. 1, according to the triple *president_of* (*Donald Trump*, *the United States*) in KB, we will label all the sentences mentioning *Donald Trump* and *the United States* as the active training instances of relation *president_of*, despite only the third sentence can express the relation correctly. What's worse, in some cases all the sentences mentioning an entity pair fail to express the relation of KB. Consequently, distant supervision strategy can easily lead to noisy labeling.

To alleviate the wrong label problem, the multi-instance learning [2] is adopted to model the task of predicting relations between entities within a bag. Concretely, a *bag* contains all the instances that mention the same entity pair e_1 and e_2. To reduce the noisy instances in a bag, the relaxed assumption [2] that *at least one* of all the sentences containing e_1 and e_2 is expressing $r(e_1, e_2)$ is applied. While this relaxed assumption is in practice often more valid than the strong one and substantially improves the extraction performance, the main weaknesses of previous methods [2]–[4] are that most features used for the relation prediction are explicitly

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Asif Naeem.

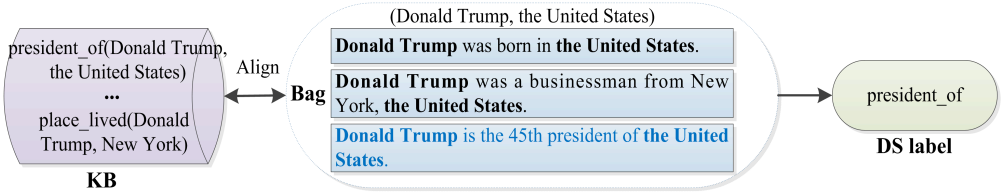


FIGURE 1. An example of distantly supervised relation extraction task.

derived from NLP tools such as POS tagging and dependency parsing, where the errors generated by NLP tools will propagate or accumulate. Especially when the length of sentence increases, the accuracy of dependency parsing decreases significantly [5]. In addition, the *at-least-once* assumption of distant supervision will lose a large amount of rich information containing in the neglected sentences as only the most likely sentences of relation candidates are used in prediction.

Due to the prominent success of deep learning, neural network is widely investigated to automatically learn features for relation extraction [6], [7] to avoid the necessity of feature handcrafting [2]–[4]. Another advantage of neural network is that it is capable of fully utilizing all the informative sentences by embedding the semantics of these sentences. Thus, the semantic prior knowledge, which is useful for relation de-noising, should be preserved as more as possible. Besides the widely used Piecewise Convolutional Neural Network (PCNN) [6], [7], Recurrent Neural Network (RNN) [8] and deep residual learning [9] are also used to learn sentence representation information. As we know, CNN allows to extract local features, and RNN is capable of extracting global features of a sequence. However, they are restricted to capture the global dependency of a long sequence. Therefore, it remains problematic to learn a good sentence representation.

To explicitly eliminate the effects of the noisy labels during model training, sentence-level attention is adopted to dynamically reduce the weights of the irrelevant sentences for a given entity pair [7], [10]. Further, recent RE work [11] begins to pay attention to label-level noise, which treats a new soft label as the ground-truth instead of the Distantly Supervised (DS) labels. The basic idea of label-level denoising is the entity pairs mentioned in similar contextual patterns tend to have similar relation labels. For example, true positive instances of relation *place_of_birth* may have some common relation patterns, like "A, born in B" and "A was born in B". Relying on the condition whether the relation mentions satisfy these contextual patterns, the noisy labels of false negatives and false positives can be corrected. However, current label-level denoising method [11] obtains a soft label by presetting static confidence for DS label, which is hard to generalize and extend. This gives us latent heuristics to explore more efficient method for label-level denoising.

To deal with these problems above, we propose a Multi-head Self-attention Network (MSNet) based label denoising method for distantly supervised relation extraction, whose framework is shown in Fig. 2. First, in order to capture the long dependency between two words without regard to their distance, we present a MSNet to learn the structural representation information of a sentence without any convolution and recurrence. Concretely, we encode the words, entities and their positions information into contextual embeddings by multi-head self-attention mechanism, then extract the prominent sentence features via max pooling operation. Empirically, we find that the proposed MSNet has a better representation performance than PCNN and BiLSTM. Second, to reduce the impact of noisy sentences in a bag, we use a basic selective self-attention mechanism to learn the bag representation, which can be fed to softmax classifier to predict the relational score of each class. Third, we propose a curriculum learning method to denoise the distantly supervised label. More specifically, we introduce a label confidence function, by which we can obtain a latent label as the ground-truth of relation classifier. The curriculum learning method of label confidence can learn the latent label of easy relation patterns firstly, then gradually incorporate them into later latent label prediction of hard patterns. This continuous learning makes the latent label prediction more consistent and reliable. Finally, the experimental results demonstrate the effectiveness and generalization of our proposed method.

Our contributions of this paper include:

- We propose a multi-head self-attention network to learn the informative sentence representation, which can capture the relatedness between two words regardless of their distance and without consider the convolutional and recurrent operations.
- We present a label confidence function to achieve the label denoising via curriculum learning. It can gradually incorporate the obtained latent label of easy relation patterns into later latent label prediction of hard patterns, which makes the latent label learning more robust.
- We evaluate the proposed method on a widely used dataset, the experimental results show the effectiveness of MSNet, and also verify the generalization of the label denoising method.

The remainder of this paper is organized as follows. Section II introduces the related work. Section III describes our proposed approach in detail. Section IV reports our

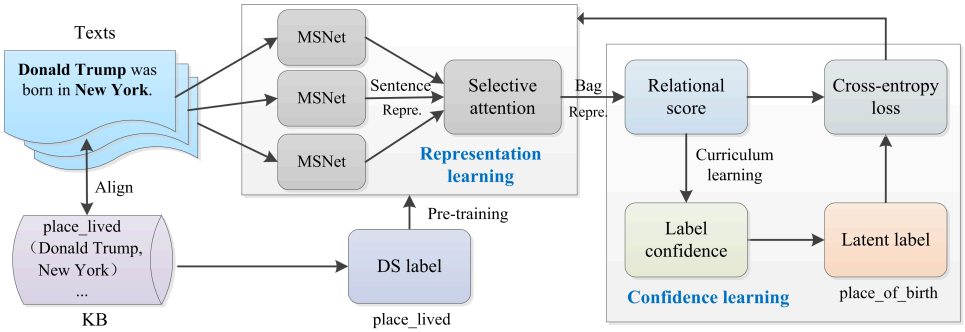


FIGURE 2. The proposed MSNet based label denoising framework for relation extraction.

experimental results, discussion and case study. Section V gives a conclusion and the future work of the whole paper.

II. RELATED WORK

In order to solve the problem of training data labeling in supervised learning, a distant supervision paradigm [1] is originally proposed to automatically create large-scale training data for RE task. However, the traditional distantly supervised method uses a strong assumption that all sentences can indicate the relation, which may easily lead to wrong labeling issue. To reduce the impact of noisy sentences in this strong assumption, multi-instance learning [2]–[4] is introduced to adopt a relaxed at-least-once assumption. This assumption holds that if there is a relation between two entities, at least one sentence that includes both entities can indicate this relation. Besides, multi-instance multi-label learning [3], [12] is employed to handle the overlapping relations problem. Nevertheless, these methods strongly rely on fine-designed features, which are costly and time-consuming. Additionally, handcrafted feature-based methods may lead to error propagation from lexical or syntactic parsing tools.

To alleviate these problems caused by feature engineering, deep learning is applied to RE task to extract features automatically. Combining multi-instance learning, Zeng *et al.* [6] employs Piecewise Convolutional Neural Network (PCNN) to train a relation extractor on distantly supervised data. The key to this method is only selecting the most likely sentence for an entity pair as a valid sentence. Then Lin *et al.* [7] assumes that a lot of information in the neglected sentences may be lost. Hence, various sentence-level attention mechanisms [7], [10] are proposed to select multiple valid sentences by assigning higher weight for valid sentences. Subsequently, reinforcement learning [13]–[15] and generative adversarial network [16] are applied to reduce noisy labeling by removing or redistributing noisy sentences. These above methods focus on sentence-level denoising and regard the relation labels from distant supervision as the ground-truth.

Recently, a soft-label denoising method [11] is introduced to use a soft label to correct the wrong labels during training, which makes full use of the contextual information from the text mentions that are correctly labeled. To obtain a soft label as the ground-truth, the method proposes a joint probability function by combining the confidence of the labels from distant supervision and the relation score from model prediction. The case study and experimental results show that this method can effectively change noisy labels during training and improve the model performance. However, the method sets a static confidence for labels obtained by distant supervision, which is inflexible and hard to extend. A recent study [17] summarizes the main factors that impact label denoising of distantly supervised RE, which indicates the prior knowledge of plain texts and partial confidence for distantly supervised labels can significantly affect the denoising performance.

Inspired by the work of Luo *et al.* [18] which utilizes curriculum learning to model the noise in the training data, we become interested in curriculum learning [19] and self-paced learning [20], [21]. The basic idea of curriculum learning is beginning with the easy examples of a task, then dealing with the hard ones. Further, self-paced learning is proposed to gradually incorporates easy to more complex examples during training. These studies give us the initial ideas about leveraging curriculum learning method for label denoising.

Self-attention [22], [23] is a special attention mechanism, which incorporates different positions of a single sequence to compute its representation. A recent breakthrough work for sequence encoder is multi-head attention [24], which achieves excellent improvement in many sequence modeling and transduction tasks [25], [26]. Compared with CNN and RNN, multi-head self-attention allows to model the inherent dependencies of a sentence regardless of the distance between two words. Moreover, multi-head attention can perform the multiple attention functions in parallel. This provides us with the basic idea to learn sentence representation via multi-head self-attention.

III. METHODOLOGY

In this section, we will introduce the proposed MSNet based label denoising framework for relation extraction. As shown in Fig. 2, the model mainly includes two modules – representation learning and confidence learning. The representation learning module can encode all the plain texts mentioning

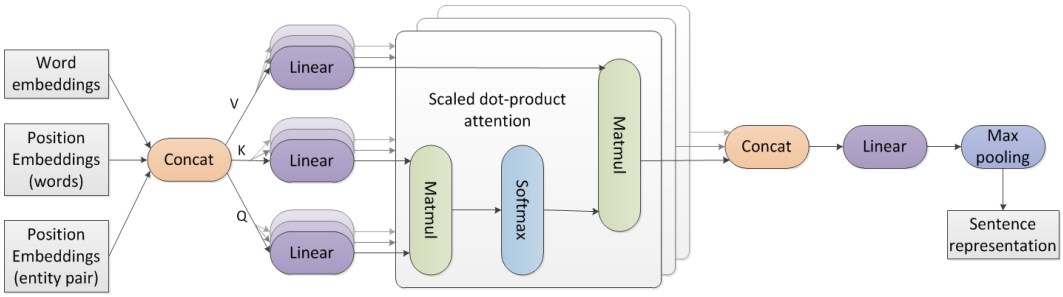


FIGURE 3. Sentence representation learning via multi-head self-attention network.

two entities into an informative representation vector which is relevant to the relation. And the confidence learning module enables the model to denoise labels by a curriculum function. The mutual learning of two modules generates a reliable latent label and effectively improves the model performance.

A. REPRESENTATION LEARNING

To be specific, representation learning is an encoding process from word level, sentence level to bag level. First, words, entities and their positions are encoded into corresponding embeddings which are machine computable. Second, these embeddings are combined to encode into a sentence representation via our proposed MSNet. Then a selective self-attention is adopted to encode all the sentence representations mentioning the same entity pair into a bag representation, which is relevant to the relation classification directly.

1) WORD AND POSITION ENCODER

Recent studies have shown that using pre-trained embeddings as input has achieved good performance in many NLP tasks such as sentence classification and relation classification [27]–[30]. To fully leverage the semantic information of a word in context, we use pre-trained *word embeddings* to map the i-th input word w_i in a sentence into a real-valued vector $\mathbf{u}_i \in \mathbb{R}^{d_w}$.

Since self-attention neglects the order of a sequence, we are required to provide the position information of each token in the sequence. Accordingly, we encode the absolute position of the words in the sentence and map the position i into a *position embedding* $\mathbf{v}_i \in \mathbb{R}^{d_p}$.

In order to capture entity information in a sentence, position features of two entities [30], [31] are used to indicate how close each word to the entity pair. More specifically, $i - pos_{e_1}$ and $i - pos_{e_2}$, the relative distances of the i-th input word in a sentence with respect to two entities at position pos_{e_1} and pos_{e_2}, are encoded with embeddings as $\mathbf{v}_{1i}, \mathbf{v}_{2i} \in \mathbb{R}^{d_p}$.

Then the word embedding and position embedding of the i-th word are concatenated with entity position embeddings as $\mathbf{w}_i = [\mathbf{u}_i, \mathbf{v}_i, \mathbf{v}_{1i}, \mathbf{v}_{2i}] \in \mathbb{R}^d (d = d_w + d_p * 3)$. Given sentence length n, the output of word-level embedding layer is combined as $\mathbf{X} = [\mathbf{w}_1, \ldots, \mathbf{w}_n] \in \mathbb{R}^{n \times d}$, which is fed to sentence encoder layer.

2) SENTENCE ENCODER

In this paper, we propose MSNet to encode the sentence representation, which combines multi-head self-attention with max pooling mechanism. Fig. 3 depicts the detailed architecture of MSNet.

Basically, we can regard an attention mechanism as a mapping of a query and key-value pairs to an output. An attention function of the query and the key is adopted to compute the weight of each value, then the output is determined as a weighted sum of the values. In particular, for self-attention, the queries, keys and values all come from the output of the previous layer $\mathbf{X} \in \mathbb{R}^{n \times d}$. As shown in Fig. 3, we use the scaled dot-product attention as the attention function. Given query matrix $\mathbf{Q}$, keys $\mathbf{K}$ and values $\mathbf{V}$, scaled dot-product attention is calculated as follows:

$$\mathcal{F}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}})\mathbf{V} \tag{1}$$

Multi-head attention captures sequence information from different semantic subspaces at different positions. First, h different linear projections are utilized to transform the query, key and value to d/h dimension respectively. Second, h projections are performed the scaled dot-product attention in parallel. Then all the outputs of h heads are concatenated and linearly projected to representation space. To be specific, multi-head attention can be formulated as below:

$$\begin{aligned} \mathbf{M} &= [\mathbf{H}_1, \mathbf{H}_2, \ldots, \mathbf{H}_h]\mathbf{W}^R \\ \mathbf{H}_i &= \mathcal{F}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V) \end{aligned} \tag{2}$$

where $\mathbf{W}_i^Q, \mathbf{W}_i^K, \mathbf{W}_i^V \in \mathbb{R}^{d \times (d/h)}$ are the parameter matrices of i-th linear projections, $\mathbf{W}^R \in \mathbb{R}^{d \times r}$ is a mapping parameter from input space to representation space, r is the dimension of sentence representation. Similar to the previous work [24], we adopt residual connection [32] around the multi-head self-attention layer followed by layer normalization [33] to improve the model performance.

In order to obtain a fixed-size vector independent of sentence length n, we use a max pooling operation for the contextual embedding $\mathbf{M} \in \mathbb{R}^{n \times r}$ by choosing the maximum value over each dimension of representation units, then form a sentence representation $\mathbf{s} \in \mathbb{R}^r$. We assume that our model can capture the most discriminative features via max

TABLE 1. Time complexity and space complexity for different network types in the sentence encoder layer. *n* is the sentence length, *d* is the dimension of input embedding, *r* is the dimension of sentence representation, *k* is the kernel size of the convolution operation.

Network Type	Time Complexity	Space Complexity
MSNet	$O(n \cdot (n \cdot d + d^2 + d \cdot r))$	$O(d^2 + d \cdot r)$
Convolution	$O(n \cdot k \cdot d \cdot r)$	$O(k \cdot d \cdot r)$
BiLSTM	$O(n \cdot (r^2 + d \cdot r))$	$O(r^2 + d \cdot r)$

pooling mechanism. In addition, we adopt the mean pooling operation to learn the sentence features by averaging the n hidden representations.

To verify the effectiveness of our proposed method, we also experiment with PCNN and BiLSTM+max-pooling to learn the sentence representation as comparison. As indicated in Table 1, with respect to different network types, we will discuss the computational complexity in the sentence encoder layer. Actually, in our experimental settings, the sentence length n almost equals the dimension of input embedding d, while the dimension of sentence representation r is much bigger than them, that is, $r \gg d \approx n$. During training, the running time of BiLSTM is about 2.3 times of MSNet and 2.8 times of PCNN. With respect to space complexity of parameters, MSNet takes less space than convolution and BiLSTM operations.

In terms of parallelized computation, we use the minimum number of sequential operation to measure it [24]. As the computation units of self-attention and convolution are separate from each other at different position in a sequence, they require constant number of sequential operations. Since the unit of next time in BiLSTM depends on the unit of previous time, BiLSTM needs $O(n)$ sequential computation.

3) BAG ENCODER

As mentioned before, our task aims to predict the relation of a bag which contains all the sentences mentioning the same entity pair. As a result, we will extract the useful sentence information from a bag with respect to an entity pair. In this layer, we use a selective self-attention method [7] to reduce noisy sentences within a bag. The attention mechanism can learn higher weight for the valid sentences and lower weight for the invalid sentences. Afterwards, the bag representation $\mathbf{o}$ of an entity pair is computed as a weighted sum of these sentence representations:

$$\mathbf{o} = \sum_i \alpha_i \mathbf{s}_i \tag{3}$$

The weight α_i of the i-th sentence representation $\mathbf{s}_i$ is computed as follows:

$$\alpha_i = \frac{\exp((\mathbf{s}_i \odot \mathbf{a})\mathbf{r})}{\sum_k \exp((\mathbf{s}_k \odot \mathbf{a})\mathbf{r})} \tag{4}$$

where $\mathbf{a}$ is a weighted vector, $\mathbf{r}$ is the query vector of the representation of relation r and $\odot$ is element-wise multiplication operation. The predicted score of the j-th relation $\mathbf{p}_j$ is computed by softmax function based on the bag representation $\mathbf{o}$.

$$\mathbf{p}_j = \frac{\exp(\mathbf{W}\mathbf{o}_j + \mathbf{b})}{\sum_k \exp(\mathbf{W}\mathbf{o}_k + \mathbf{b})} \tag{5}$$

where $\mathbf{W}$ and $\mathbf{b}$ are training parameters which indicate the relation matrix and bias vector respectively.

B. CONFIDENCE LEARNING

To perform label-level denoising by learning a latent label as the ground-truth, we hope to make full use of DS labels because the overwhelming majority of labels from distant supervision are correct. In addition to distantly supervised information, the semantic patterns captured by relation extractor also provide sufficient information for the generation of latent labels. Accordingly, the primary problem of our label denoising is to design an efficient latent-label learning algorithm that can dynamically weigh the DS label and relation pattern.

Inspired by curriculum learning [19]–[21], we will process the relation patterns in a meaningful order from easy to hard. More specifically, these easy examples of common relational patterns will be handled by our model firstly, then the hard examples with special patterns will be processed based on the information from previous examples. Intuitively, if the relation extractor has a good performance, it will predict higher scores for the correct labels of the easy patterns than the hard patterns. Instead of manually splitting the examples, we plan to leverage what the model has learned from the DS labels to handle the examples with different predicted scores from high to low.

We will describe the proposed latent-label learning algorithm, which is an extension of dynamic soft label [17]. Initially, to obtain prediction power of relation patterns, we use DS labels to pre-train the representation learning module. With the pre-training proceeding, model performance fails to be further improved when there is no more new information of distant supervision to learn. We call the time-step that our model has a stable predictive power as the boundary between DS-label pre-training and latent-label training. Once the model training reaches the boundary, we start to use the latent label to train representation learning module. Concretely, we model the generation of latent label based on its confidence in different training steps. Given the number of labels K, the label confidence $\mathbf{c}^{(\tau)} \in \mathbb{R}^K$ represents a score vector corresponding to K relation classes in the τ-th training step. The latent label $\hat{y}^{(\tau)}$ in the τ-th training step can be computed as below:

$$\begin{aligned} \hat{y}^{(\tau)} &= \arg\max \{\mathbf{c}_1^{(\tau)}, \mathbf{c}_2^{(\tau)}, ..., \mathbf{c}_K^{(\tau)}\} \\ \mathbf{c}^{(\tau)} &= \mathcal{G}(\mathbf{c}^{(\tau-1)}, \mathbf{p}^{(\tau)}) \end{aligned} \tag{6}$$

where $\mathcal{G}(\cdot)$ represents the curriculum function, through which we can not only utilize the predicted score $\mathbf{p}^{(\tau)}$ in current training step as mentioned before, but also preserve the label confidence $\mathbf{c}^{(\tau-1)}$ in preceding training step.

In order to model the impact of latent labels in the previous training step, we consider the correlation between label confidences in adjacent training steps in this curriculum function. In addition, the information of relation patterns should be included to guide the continuous learning of model prediction. Hence, we design the curriculum function which combines preceding confidence and current model prediction linearly as follows:

$$\begin{aligned} \mathbf{z}_k^{(\tau)} &= \beta^{(\tau)} * \mathbf{c}_k^{(\tau-1)} + (1-\beta^{(\tau)}) * \mathbf{p}_k^{(\tau)} \\ \mathbf{c}_j^{(\tau)} &= \frac{\mathbf{z}_j^{(\tau)}}{\sum_{k=1}^{K} \mathbf{z}_k^{(\tau)}} \end{aligned} \tag{7}$$

where $\beta^{(\tau)} \in (0, 1]$ is a weight factor in the τ-th training step that weighs the label confidence from preceding prediction and relational score from current model prediction. It is calculated by $\beta^{(\tau)} = \beta^{(\tau-1)} + \delta$, where $\delta \in (0, 1)$ is a delay factor that measures the difference between two weight factors in the adjacent step and accelerates model convergence. In particular, if $\beta^{(\tau)} = 1$, our model will always use the DS label as the ground-truth label. To ensure that each value of label confidence $\mathbf{c}^{(\tau)}$ varies in the range (0,1), the obtained vector $\mathbf{z}^{(\tau)}$ by weighted sum is normalized over all K classes. And we set the initial confidence $\mathbf{c}^{(0)}$ as the one-hot vector of the DS label.

In general, the iterative learning can be viewed as a process of seeking the optimal point that trades off the information of distant supervision and model prediction. With the memory of preceding confidence, the model can preserve the information of distant supervision. As the training proceeds, the effect of model prediction enhances and more relation patterns are learned. More concretely, due to the high predicted scores of easy patterns, their latent labels are likely to retain the same as DS labels or change to other labels in the front iterations. Based on the learning of these easy patterns, the latent labels of hard patterns may gradually change in later iterations. Thus our curriculum learning of label confidence can generate latent labels in a reliable way from easy patterns to hard patterns.

C. TRAINING

During training, we will optimize the entire neural network by minimizing the objective function. In this paper, we use cross-entropy loss as the objective function. In order to obtain stable model prediction ability, we need to pre-train our model via distantly supervised information. Let $\mathcal{T}$ be the boundary of training steps which represents whether our model can stably predict the relation patterns. At the beginning of training steps $\tau <= \mathcal{T}$, the DS label y is employed as the ground-truth label, the objective function is computed as follows:

$$\mathcal{L}(\mathbf{y}, \mathbf{p}) = -\sum_{j=1}^{K} (\mathbf{y}_j \log \mathbf{p}_j) \tag{8}$$

where $\mathbf{p}$ is the relation probability of model prediction, and $\mathbf{y}$ is the one-hot vector of the DS label. Then when $\tau > \mathcal{T}$ the latent label $\hat{y}$ is regarded as the gold label, the objective function is changed as below:

$$\mathcal{L}(\hat{\mathbf{y}}, \mathbf{p}) = -\sum_{j=1}^{K} (\hat{\mathbf{y}}_j \log \mathbf{p}_j) \tag{9}$$

Algorithm 1 Model Training

Input: Training set $\mathcal{D}$, training boundary $\mathcal{T}$
1 Initialize network parameters θ
2 **for** *epoch* $\tau = 1$ *to* N **do**
3 **if** $\tau <= \mathcal{T}$ **then**
4 Predict the current relational score $\mathbf{p}^{(\tau)}$ by (5)
5 Update parameters with DS label via $\mathcal{L}(\mathbf{y}, \mathbf{p}^{(\tau)})$
6 **else**
7 Compute label confidence $\mathbf{c}^{(\tau)}$ by (7)
8 Obtain latent label $\hat{\mathbf{y}}^{(\tau)}$ by (6)
9 Predict the current relational score $\mathbf{p}^{(\tau)}$ by (5)
10 Update parameters with latent label via $\mathcal{L}(\hat{\mathbf{y}}^{(\tau)}, \mathbf{p}^{(\tau)})$
11 **end**
12 **end**

The model training procedure is described in Algorithm 1. Concretely, the entire training process is divided into two parts through the boundary $\mathcal{T}$ of training steps. When $\tau <= \mathcal{T}$, we will pre-train the representation learning module via DS label in lines 4-5, which aims to get stable prediction ability of relation classifier. When $\tau > \mathcal{T}$, we will use the proposed label denoising method to obtain a latent label in lines 7-8, and utilize this label to train the relation classifier in lines 9-10. All the network parameters are updated via mini-batch Adam optimizer [34].

IV. EXPERIMENTS

In this section, the performance of the proposed method is evaluated and compared with the state-of-the-art RE systems. Additionally, we present the effects of different learning modules. Finally, the case study demonstrates that our approach can gradually correct the noisy labels effectively.

A. DATASET AND EVALUATION

Our method is evaluated on a popular benchmark dataset developed by Riedel *et al.* [2]. The dataset was generated by aligning relational triples of Freebase with the New York Times (NYT) corpus, and was widely used by various methods for distantly supervised RE task. It has 52 relation classes and a NA class which indicates non-relation between entity pairs. The training set contains 522,611 sentences, 281,270 entity pairs and 18,252 relation triples. The test set contains 172,448 sentences, 96,678 entity pairs and 1,950 relation triples. Following previous work [7], [11], [18], we employ the held-out evaluation, which compares the predicted labels between entity pairs with the DS labels without any human evaluation. Our model is evaluated with

aggregate precision-recall curves and top N Precision (P@N) for relation extraction with different number of sentences between entity pairs.

B. EXPERIMENTAL SETTINGS

We use Tensorflow 1.4.0 and complete the code with Python 3.5. All experiments are conducted on a single Nvidia Titan X GPU. Similar to prior work [6], [7], we use cross-validation to identify the parameters of this paper. For fair comparison, we adopt the same pre-trained word embeddings released by Lin *et al.* [7] which have $d_w = 50$ dimensions and are trained on this dataset. We choose the dimension of position embeddings $d_p = 5$, the number of heads $h = 5$ and the number of hidden unit $r = 230$ for our MSNet. Additionally, the window size and the number of feature maps for PCNN model are set to 3 and 230, and the number of hidden units for BiLSTM is also set to 230. We assign the initial weight factor $\beta^{(0)} = 0.5$ and the delay factor $\delta = 0.1$ in confidence learning module. To avoid overfitting, dropout [35] is applied to sentence representation. Concretely, we use dropout rate 0.5 and l_2 constraint 0.0001. Moreover, we use Adam optimizer [34] with learning rate 0.001 and batch size 50. Following the experimental observations, we find the model performs relatively stable on the precision-recall curve after around 3 training epochs, which indicates there is no additional information we can learn from the DS label. To fully learn the contextual patterns of relation mentions at the beginning of training, we utilize latent label after 5 epochs.

C. RESULTS AND DISCUSSION

In order to assess our proposed approach, we compare it with the various methods. Mintz *et al.* [1] is a traditional distantly supervised method which extracts features from all the instances. MultiR [3] and MIMLRE [12] are multi-instance learning and multi-instance multi-labels methods respectively. These three models are handcrafted feature-based methods, all the results come from their reported papers. PCNN [7] with selective attention model serves as the baseline of the neural network based methods, and PCNN+soft-label [11] is a label-level denoising method which sets a static confidence for DS label. We reproduce the results of PCNN and PCNN+soft-label according to their settings, which may be slightly different from their reported results.

Fig. 4 shows the precision-recall curves of our proposed model and the compared methods. Following previous work [7], [11], we only report the top-ranked results when the recall is less than 0.4. From the figure, we can observe that: (1) Neural network based methods [7], [11] obviously achieve higher precision than handcrafted feature based methods [1], [3], [12]. This demonstrates the handcrafted feature based methods are problematic especially when there are noisy labels in the task. (2) MSNet achieves higher precision than the PCNN [7] baseline, which verifies the excellent learning ability of MSNet. (3) Adding Latent-label mechanism for MSNet or adding Soft-label [11] for PCNN can notably improve the performance of baselines, which indicates the effectiveness of using label denoising. (4) The proposed MSNet+Latent-label presents the best performance compared with all previous methods, and achieves the highest precision in almost the entire range when the recall is greater than 0.05. Actually, we find that there are many wrong labeled entity pairs by manual evaluation when recall is less than 0.05 like the work [11]. In particular, our model slightly outperforms PCNN+soft-label to some extent. We assume that the design of static confidence of soft-label method [11] considers the class imbalance problem, while our model ignores it in order to generalize the proposed method to other distantly supervised tasks. As a whole, our model significantly performs better than the compared methods.

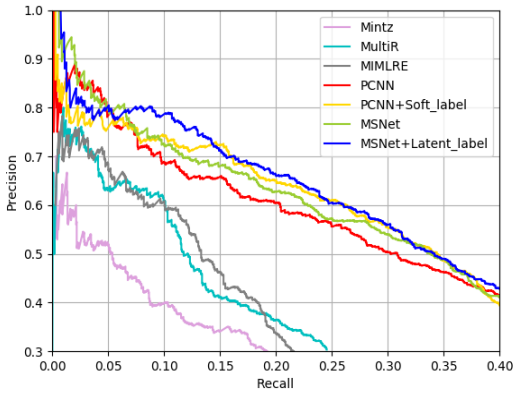


FIGURE 4. Performance comparison of different methods with precision-recall curves.

TABLE 2. Precision of various models for different recalls.

Method	Recall			
Precision (%)	0.1	0.2	0.3	0.4
Mintz [1]	39.9	28.6	16.8	-
MultiR [3]	60.9	36.4	-	-
MIMLRE [12]	60.7	33.8	-	-
PCNN	68.7	60.5	50.4	41.4
+soft-label [11]	74.1	65.1	55.4	39.5
+latent-label	74.4	65.3	53.3	43.2
BiLSTM	75	59.0	47.9	37.1
+soft-label [11]	60.7	45.4	30.8	7.9
+latent-label	76.5	65.5	53.6	41.4
MSNet(Mean pooling)	69.1	58.3	49.7	38.5
+soft-label [11]	72.2	58.8	47.9	35.4
+latent-label	71.7	66.2	53.6	41.7
MSNet(Max pooling)	72.5	62.8	54.0	41.2
+soft-label [11]	70.1	59.8	44.8	30.9
+latent-label	**78.6**	**66.3**	**55.9**	**42.9**

To give a clear description, Table 2 demonstrates the precision of a variety of models for different recalls 0.1/0.2/0.3/0.4. And we highlight the best results with bold formatting. We can find that combining latent-label mechanism improves the performance remarkably for MSNet, PCNN and BiLSTM respectively, which significantly exceeds the soft-label method [11]. In particular, soft-label method only improves the PCNN baseline when recall is less than 0.3 while fails to improve MSNet and BiLSTM, which reflects the instability of soft-label denoising method. Moreover, we replace the

TABLE 3. Top N precision for relation extraction in the entity pairs with different number of sentences.

Settings	One				Two				All			
P@N(%)	100	200	300	Mean	100	200	300	Mean	100	200	300	Mean
PCNN	74	70	65	69.7	78	72	68.3	72.8	78	70	70.3	72.8
+ soft-label [11]	76	72	68	72	80	76	69.7	75.2	81	77	71.7	76.6
+ latent-label	77	71	68	72	82	76.5	**73.7**	77.4	85	79	73.7	79.2
BiLSTM	80	68.5	60.7	69.7	80	70.5	64	71.5	83	74.5	67	74.8
+ soft-label [11]	68	52.5	46.3	55.6	74	62	55.7	63.9	79	67	58.7	68.2
+ latent-label	80	72	67.3	73.1	84	**80**	71.3	78.4	85	78.5	73.7	79.1
MSNet	76	70.5	65	70.5	77	73	68	72.7	81	74	71.3	75.4
+ soft-label [11]	78	67	61.7	68.9	83	73.5	63.3	73.3	83	76.5	69	76.2
+ latent-label	**83**	**73.5**	**68.3**	**74.9**	**89**	**80**	71	**80**	**88**	**82.5**	**76**	**82.2**

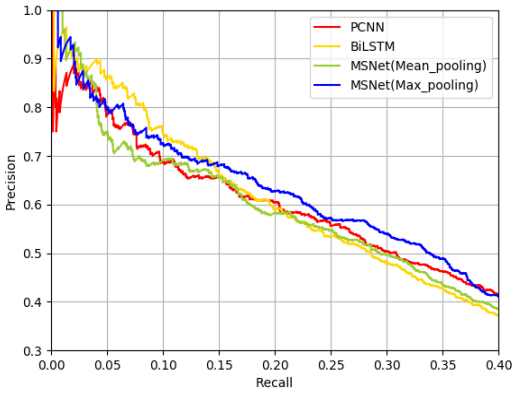


FIGURE 5. Performance comparison of different representation learning modules.

max pooling operation with mean pooling in MSNet, and we observe that combining with latent-label also makes it achieve higher precision than its corresponding baseline. Therefore, our latent-label denoising method can generalize to different representation learning networks effectively.

Following the evaluations of previous work [7], [11], we randomly select one, two and all sentences from the entity pairs which have more than one sentence to report the top N precision (P@N). Table 3 presents the results of different settings for top 100/200/300 and their mean value in test data. It is clear that adding latent-label mechanism performs well for all the neural baselines, which indicates its effectiveness when combining with sentence-level denoising. In particular, our proposed model almost remains the best performance in the entire P@N range.

1) EFFECT OF REPRESENTATION LEARNING MODULE

Fig. 5 depicts the precision-recall curves of representation learning modules constructed by different neural networks - MSNet, PCNN and BiLSTM respectively. At this time, we use DS labels during training and ignore the effect of label denoising mechanism. Compared with the widely used PCNN model, we observe that BiLSTM performs better when the recall is less than 0.2, but it drops sharply with recall greater than 0.2. The MSNet outperforms PCNN on the whole and has a relatively stable performance in the entire range. Furthermore, we replace the max pooling operation in MSNet with mean pooling operation and find that it makes a worse performance than PCNN baseline. Since we obtain contextual embeddings via multi-head self-attention method, we assume that max pooling operation can make the model get more discriminative features for a sentence than mean pooling operation. Thus, the results prove the effectiveness of our proposed MSNet for sentence representation learning.

2) EFFECT OF CONFIDENCE LEARNING MODULE

As shown in Fig. 6, to verify the effect of our proposed confidence learning for label-level denoising, we analyze the precision-recall curves for four representation learning modules above. FromFig. 6(a) and Fig. 6(b), we can discover that latent-label can notably improve MSNet and MSNet with mean pooling baselines. Nevertheless, soft-label method [11] drops rapidly in MSNet when recall is greater than 0.15 in Fig. 6(a), and remains nearly equal to the MSNet with mean pooling baseline in Fig. 6(b). From Fig. 6(c), we find that soft-label method even gets a slightly better performance than our proposed latent-label method of the recall range from 0.2 to 0.35. However, as illustrated in Fig. 6(d), the soft-label method has a very poor performance in BiLSTM while latent-label method can achieve higher precision when the recall is greater than 0.1. Clearly, the latent-label scheme via our confidence learning algorithms has better generalization than the soft-label scheme. In the following, we will examine the effects of curriculum learning by case study in detail.

D. CASE STUDY

We show the visualization of attention weight in Fig. 7 to verify the effectiveness of selective self-attention mechanism. And we use a darker color to represent higher weight of the corresponding sentence in a bag. With respect to the relation *place_of_death*, the second sentence is arranged higher attention weight since it indicates that *Kurt Waldheim* was died in *Vienna*, while the first sentence which shows that *Kurt Waldheim* was born in *Vienna* gets very low weight. For another relation *contains*, the second and fourth sentences have higher attention weight than other sentences because they can clearly express the relation between locations in a more common pattern. Therefore, the selective self-attention mechanism is capable of learning higher weight for more valid sentences.

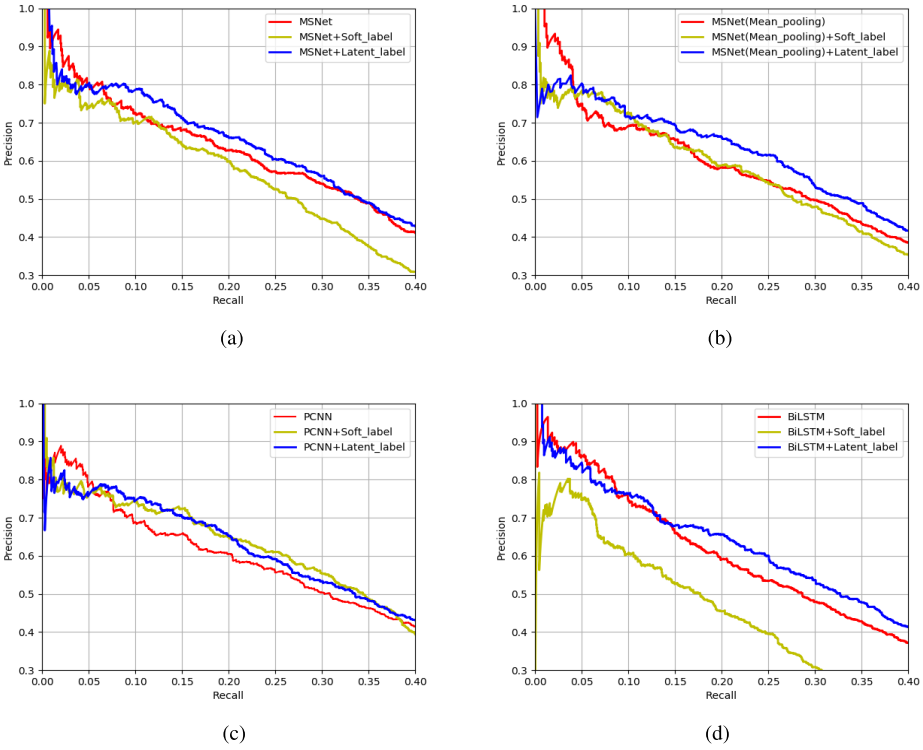


FIGURE 6. Effects of confidence learning for four representation learning modules. (a) MSNet. (b) MSNet(Mean pooling). (c) PCNN. (d) BiLSTM.

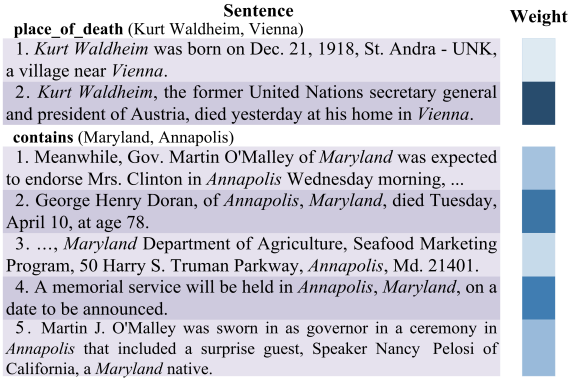


FIGURE 7. Visualization of selective self-attention weight.

Then we will discuss typical cases of latent label correction in detail. We manually check the label corrections of 200 random relational bags. Correction accuracy of the proposed label denoising method is 88%(176/200), which proves our model can correct the noisy labels with high accuracy. In addition, we present some cases of latent-label corrections to test our proposed curriculum confidence learning, which are shown in Table 4. Note that we highlight the accurate labels with bold formatting.

We present some right label corrections in the upper part of Table 4. We find the noisy labels of false negatives can be corrected by latent labels. For instance, we correctly recognize the relation *company* between *Helmut Panke* and *BMW*, whose relationship is missed in Freebase. Moreover, we can correct some false positives by latent labels, such as DS labels *nationality* and *place_of_birth*. Relational triple *nationality*(*Didier Drogba, France*) of Freebase is obviously mislabeled, because *Didier Drogba* was born in Ivory Coast. For entity pair (*Pericles, Athens*), their relation is labeled as *place_of_birth* while the instance indicates that *Pericles* died in *Athens*. Through the learning of patterns, we can correct it with latent label *place_of_death*. These relation patterns can be regarded as easy patterns, whose noisy labels can be directly corrected by first several training epochs. The last example of right corrections also fails to express the DS labels *place_lived* between two entities *Rem Koolhaas* and *Rotterdam*. Fortunately, we observe that curriculum learning mechanism makes gradual latent-label correction. At the beginning, the latent label is corrected as *place_of_birth*, finally it is corrected as NA. It indicates that noisy labels of the examples with hard patterns can be corrected effectively based on curriculum learning.

We also show the cases of wrong corrections in the lower part of Table 4. For the first example, our model wrongly corrects the relation between two entities *Kapolei* and *Honolulu* as NA, because its relational pattern is different from the common patterns of relation *contains* such as "*A in B*" or "*A, B*". With respect to the second example, our model may just discover that relation pattern "*A, co-chairman of B*" corresponds to relation *company*, but mistakenly assume that entity *Glendale* is a company name. Due to the diversity of relation patterns and limitations of model capabilities,

TABLE 4. Case study of latent-label correction.

Type	DS label	Latent label	Instances
	NA	**company**	*Helmut Panke* of *BMW* is one of the most successful auto chief executives in Europe.
Right	nationality	**NA**	*Didier Drogba* joined Chelsea in may 2004 from Olympique Marseille of *France* for almost $50 million.
corrections	place_of_birth	**place_of_death**	A majority were killed or executed on campaign, died of wounds, or (like *Pericles*) of the plague that swept through overcrowded *Athens*, ...
	place_lived	place_of_birth→ **NA**	..., it may have seemed a comedown after working together for years at OMA, *Rem Koolhaas*'s big firm in *Rotterdam*.
Wrong	**contains**	NA	*Kapolei* is on the southern coast of Oahu, about 20 miles from *Honolulu*.
corrections	**NA**	company	..., said *Bran Ferren*, a former Disney studios designer and technologist, who is now co-chairman of applied minds, a technology consulting firm based in *Glendale*, Calif.

we think that a small amount of mistakes are inevitable. As a whole, our proposed method is very effective for correcting noisy labels and improving RE performance.

V. CONCLUSION

In this paper, we propose a MSNet based label denoising method for distantly supervised relation extraction. We introduce a multi-head self-attention mechanism to learn the sentence representation without any convolutional and recurrent operations. Moreover, we adopt a label confidence curriculum learning method to correct the noisy labels gradually. The experimental results show that our approach can outperform the state-of-the-art methods significantly. With respect to sentence representation learning, our proposed MSNet gets a better performance than PCNN and BiLSTM based model. Curriculum learning of label confidence can gradually incorporate the obtained latent label of easy relation patterns into later latent label prediction of hard patterns, and rectify the noisy labels with high accuracy. In addition, the latent-label denoising mechanism can generalize to PCNN and BiLSTM respectively.

In the future, we will plan to explore the label-level denoising method via reinforcement learning and generative adversarial network. Further, we consider to solve the multi-label denoising problem by modeling the relatedness between labels.

REFERENCES

[1] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, "Distant supervision for relation extraction without labeled data," in *Proc. Joint Conf. 47th Annu. Meeting ACL 4th Int. Joint Conf. Natural Lang. Process. (AFNLP) Assoc. Comput. Linguistics*, Aug. 2009, pp. 1003–1011.

[2] S. Riedel, L. Yao, and A. McCallum, "Modeling relations and their mentions without labeled text," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Berlin, Germany: Springer, 2010, pp. 148–163.

[3] R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, and D. S. Weld, "Knowledge-based weak supervision for information extraction of overlapping relations," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics, Hum. Lang. Technol.*, Jun. 2011, pp. 541–550.

[4] A. Ritter, L. Zettlemoyer, and O. Etzioni, "Modeling missing data in distant supervision for information extraction," *Trans. Assoc. Comput. Linguistics*, vol. 1, pp. 367–378, Oct. 2013.

[5] R. McDonald and J. Nivre, "Characterizing the errors of data-driven dependency parsing models," in *Proc. EMNLP-CoNLL*, 2007, pp. 122–131.

[6] D. Zeng, K. Liu, Y. Chen, and J. Zhao, "Distant supervision for relation extraction via piecewise convolutional neural networks," in *Proc. EMNLP*, 2015, pp. 1753–1762.

[7] Y. Lin, S. Shen, Z. Liu, H. Luan, and M. Sun, "Neural relation extraction with selective attention over instances," in *Proc. ACL*, vol. 1, 2016, pp. 2124–2133.

[8] Y. Wu, D. Bamman, and S. Russell, "Adversarial training for relation extraction," in *Proc. EMNLP*, 2017, pp. 1778–1783.

[9] Y. Y. Huang and W. Y. Wang, "Deep residual learning for weakly-supervised relation extraction," in *Proc. EMNLP*, 2017, pp. 1803–1807.

[10] G. Ji, K. Liu, S. He, and J. Zhao, "Distant supervision for relation extraction with sentence-level attention and entity descriptions," in *Proc. 31st AAAI Conf. Artif. Intell.*, Feb. 2017, pp. 3060–3066.

[11] T. Liu, K. Wang, B. Chang, and Z. Sui, "A soft-label method for noise-tolerant distantly supervised relation extraction," in *Proc. EMNLP*, 2017, pp. 1791–1796.

[12] M. Surdeanu, J. Tibshirani, R. Nallapati, and C. D. Manning, "Multi-instance multi-label learning for relation extraction," in *Proc. Joint Conf. Empirical Methods Natural Lang. Process. Comput. Natural Lang. Learn. Assoc. Comput. Linguistics*, Jul. 2012, pp. 455–465.

[13] X. Zeng, S. He, K. Liu, and J. Zhao, "Large scaled relation extraction with reinforcement learning," in *Proc. AAAI*, 2018, pp. 5658–5665.

[14] J. Feng, M. Huang, L. Zhao, Y. Yang, and X. Zhu, "Reinforcement learning for relation classification from noisy data," in *Proc. AAAI*, 2018, pp. 1–8.

[15] P. Qin, W. Xu, and W. Y. Wang, "Robust distant supervision relation extraction via deep reinforcement learning," in *Proc. ACL*, 2018, pp. 2137–2147.

[16] P. Qin, W. Xu, and W. Y. Wang, "DSGAN: Generative adversarial training for distant supervision relation extraction," in *Proc. ACL*, 2018, pp. 496–505.

[17] T. Sun, C. Zhang, and Y. Ji, "Factors impacting the label denoising of neural relation extraction," in *Proc. 12th Int. Conf. Algorithmic Aspects Inf. Manage. (AAIM)*. Cham, Switzerland: Springer, 2018, pp. 12–23.

[18] B. Luo *et al.*, "Learning with noise: Enhance distantly supervised relation extraction with dynamic transition matrix," in *Proc. ACL*, 2017, pp. 430–439.

[19] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, Jun. 2009, pp. 41–48.

[20] M. P. Kumar, B. Packer, and D. Koller, "Self-paced learning for latent variable models," in *Proc. 23rd Int. Conf. Neural Inf. Process. Syst.*, Dec. 2010, pp. 1189–1197.

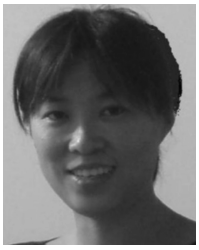
[21] L. Jiang, D. Meng, Q. Zhao, S. Shan, and A. G. Hauptmann, "Self-paced curriculum learning," in *Proc. AAAI*, 2015, pp. 2694–2700.

[22] Y. Liu, C. Sun, L. Lin, and X. Wang. (2016). "Learning natural language inference using bidirectional lstm model and inner-attention." [Online]. Available: https://arxiv.org/abs/1605.09090

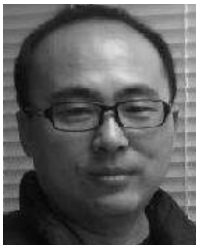
[23] Z. Lin *et al.* (2017). "A structured self-attentive sentence embedding." [Online]. Available: https://arxiv.org/abs/1703.03130

[24] A. Vaswani *et al.*, "Attention is all you need," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, Dec. 2017, pp. 6000–6010.
[25] Z. Tan, M. Wang, J. Xie, Y. Chen, and X. Shi, "Deep semantic role labeling with self-attention," in *Proc. AAAI*, 2018, pp. 1–8.
[26] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. (2018). "Bert: Pre-training of deep bidirectional transformers for language understanding." [Online]. Available: https://arxiv.org/abs/1810.04805
[27] T. Mikolov, W.-T. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations," in *Proc. HLT-NAACL*, vol. 13, 2013, pp. 746–751.
[28] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. EMNLP*, vol. 14, 2014, pp. 1532–1543.
[29] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. EMNLP*, 2014, pp. 1746–1751.
[30] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, "Relation classification via convolutional deep neural network," in *Proc. COLING*, 2014, pp. 2335–2344.
[31] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *J. Mach. Learn. Res.*, vol. 12, pp. 2493–2537, Aug. 2011.
[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[33] J. L. Ba, J. R. Kiros, and G. E. Hinton. (2016). "Layer normalization." [Online]. Available: https://arxiv.org/abs/1607.06450
[34] D. P. Kingma and J. Ba. (2014). "Adam: A method for stochastic optimization." [Online]. Available: https://arxiv.org/abs/1412.6980
[35] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

TINGTING SUN received the M.S. degree in electronic and communication engineering from the Beijing University of Posts and Telecommunications (BUPT), in 2015, where she is currently pursuing the Ph.D. degree with the School of Information and Communication Engineering. Her research interests include natural language processing, deep learning, and information extraction.



CHUNHONG ZHANG received the B.Eng. and M.S. degrees in information technology, and the Ph.D. degree in computer science from the Beijing University of Posts and Telecommunications (BUPT), in 1993, 1996, and 2013, respectively. She was a Visiting Scholar with the Illinois Institute of Technology, in 2015. She is currently a Lecturer with the School of Information and Communication Engineering, BUPT. Her research has been supported by the Ministry of Science and Technology of China and the National Natural Science Foundation of China. Her research interests include deep learning, ubiquitous computing, and data mining.



YANG JI received the Ph.D. degree in electronic engineering from the Beijing University of Posts and Telecommunications (BUPT), in 2002, where he is currently a Professor. He has conducted a lot of significant projects with domestic industry and academic partners with the support from the Ministry of Science and Technology and the Ministry of Industry and Information Technology of China, and he also had a lot of cooperation with international partners with the support from the EU Commission in the past years. His research interests include ubiquitous computing, Web information systems, and network technology.

ZHENG HU received the B.S. degree from the Nanjing University of Posts and Telecommunications, in 2002, and the Ph.D. degree in circuits and systems from the Beijing University of Posts and Telecommunications (BUPT), Wuxi, China, in 2008, where he is currently with the State Key Laboratory of Networking and Switching Technology, and he is also with the Institute of Sensing Technology and Business. He is involved in ubiquitous networking and service computing. His current interests include user behavior modeling and analysis in the mobile Internet, and social networks.