This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2930340, IEEE Access

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Reinforcement Learning for Distantly Supervised Relation Extraction

TINGTING SUN[1], CHUNHONG ZHANG[1], YANG JI[1] AND ZHENG HU[1]

[1]State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

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

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

ABSTRACT Relation extraction is a task of identifying semantic relations between entity pairs from plain text, which can benefit a lot of AI applications such as knowledge base construction and answer questioning. The distant supervision strategy is introduced to automatically create large-scale training data, which inevitably suffers from noisy label problem. Recent works handle the sentence-level denoising by reinforcement learning, which regards the labels from distant supervision as the ground-truth. However, few works focus on the label-level denoising that corrects noisy labels directly. In this paper, we propose a reinforcement learning based label denoising method for distantly supervised relation extraction. The model consists of two modules: extraction network (ENet) and policy network (PNet). The core of our label denoising is designing a policy in the PNet to obtain latent labels, where we can select the actions of using the distantly supervised labels or the predicted labels from the ENet. More concretely, the task can be modeled as an iterative process. First, the ENet predicts the relation probability, through which the model generates state representation. Second, the PNet learns the latent labels with taken actions and uses them to update the ENet. Then the optimized ENet gives the rewards to the PNet. The joint learning of two modules can obtain a reliable latent label and effectively improve the classification performance. Experimental results show that reinforcement learning is effective for noisy label correction and the proposed method can outperform the state-of-the-art relation extraction systems.

INDEX TERMS Relation extraction, distant supervision, reinforcement learning, noisy label

I. INTRODUCTION

Relation extraction is an important research branch in the field of Natural Language Processing (NLP), which aims to recognize semantic relationship between entity pairs from unstructured text. The extracted relational facts are crucial for a variety of Artificial Intelligence (AI) tasks including information extraction [1], knowledge base construction [2], [3] and question answering [4], [5]. Supervised relation extraction approaches are costly to create a great amount of training data, which is very time-consuming and labor intensive. Then distant supervision paradigm [6] for relation extraction task is proposed to automatically obtain large-scale training data by aligning the triples in Knowledge Bases (KBs) with the sentences in the text corpus. Concretely, if there is a relation between an entity pair in KBs, the basic assumption of distant supervision is that all sentences containing the given entity pair can express the relation in KBs to some extent. Nevertheless, the strong assumption seriously suffers from wrong labeling problem, especially when the KB and text are not directly related. As shown in Figure 1, the sentences that mention the entity pair *Donald Trump* and *U.S.* are labeled as the training instances of the relation *President_of*, although none of the sentences can express this relation, thus we regard it as a noisy label.

In order to mitigate the problem caused by strong assumption, a relaxed expressed-at-least-once assumption [7] is introduced, which can be seen as a multi-instance learning strategy. More specifically, the assumption is that at least one sentence referring to two entities might express the relation in KB, which is proven to take effect for alleviating wrong labeling issue. However, earlier methods [7]–[9] heavily rely on well-designed handcrafted features that require the involvement of domain experts. Additionally, most of the features are derived from NLP tools, which may lead to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2930340, IEEE Access

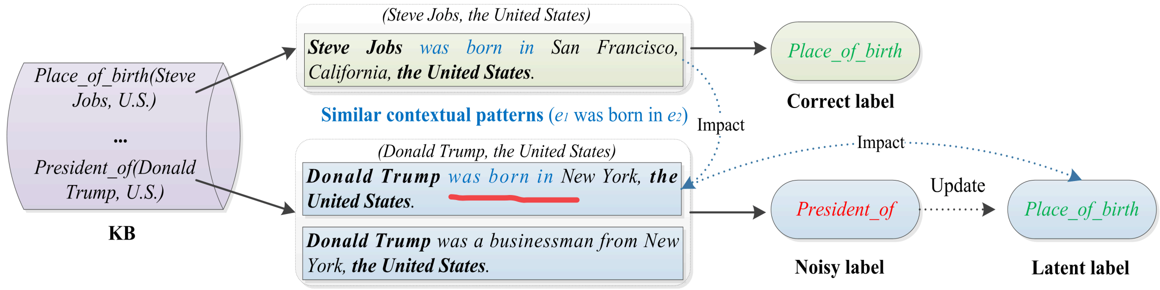


FIGURE 1: An example of label denoising for distantly supervised relation extraction.

the propagation and accumulation of errors. Owing to the successful application of deep learning in AI tasks, neural network [10] is employed to automatically extract features for relation extraction task. To make full use of the rich information contained in the neglected sentences of expressed-at-least-once assumption, sentence attention [11], [12] is adopted to arrange higher weight for more valid sentences.

Except for sentence noise above, the researchers also make great efforts to handle label noise [13]. Existing label denoising methods [13]–[15] correct noisy labels generated from distant supervision by treating a new label as the ground-truth, which is obtained by trading off the relational score from model prediction and distantly supervised information. The basic premise of label denoising is to predict relation labels for entity pairs with similar contextual patterns in text corpus. As illustrated in Figure 1, through the training of relation extractor, the model may discover that a common pattern of relation *Place_of_birth* is “A was born in B”, so noisy label *President_of* of the second example can be impacted by true positives like the first example and updated to a correct label *Place_of_birth*. Hence the noisy labels of false negatives and false positives can be corrected taken into account their relation patterns.

More recently, deep Reinforcement Learning (RL) [16], which adopts deep learning algorithms in RL, has been used successfully to denoise sentences for distantly supervised relation extraction task. One solution of prior works [17], [18] is to remove noisy sentences with RL and select the high-quality sentences to train the relation classifier. Another solution [19] is to automatically recognize the false positives with RL, and redistribute the false positives into the negative examples. Whereas, these methods still focus on the sentence denoising and regard noisy labels as the gold standard. Further, we intend to correct noisy labels with appropriate state representations and rewards in RL network. These intuitions provide a potential inspiration for us to design a more flexible and efficient RL algorithm for label denoising.

In this paper, we creatively adopt RL to achieve label denoising for distantly supervised relation extraction. The innovation points of our work include: 1) Our RL method is based on label denoising that improves relation extraction performance by reducing noisy labels, while previous RL methods [17]–[19] improve the performance by reducing noisy sentences. 2) Our label denoising method uses policy function to learn the interaction between semantic information and distantly supervised labels, while prior label denoising methods [13]–[15] adopt simple linear function. Therefore, our work is innovative compared with all the previous works. The proposed framework is shown in Figure 2. At the beginning, we employ a typical extraction network to obtain the relation score. More specifically, information of words and entity positions is encoded into sentence embeddings by neural network, then a common selective attention is utilized to choose more valid sentences to predict the score for each relation. The information of the predicted score and the labels from distant supervision are combined as input states of RL. Then we adopt the proposed policy network to generate latent labels. In the policy network, we design a policy to get the binary actions that determine distantly supervised labels or predicted labels as latent labels. Afterwards, we use obtained latent labels to supervise the training of extraction network, which can provide rewards to policy network and the input states of the next RL episode. This joint learning of extraction network and policy network makes the latent label prediction reliable and significantly improve the performance of relation extraction.

Our contributions of this paper are to:

- propose a RL based label denoising method for distantly supervised relation extraction. The framework comprises two modules: extraction network and policy network. Joint training of two modules can correct the noisy label without any extra supervision.
- design a policy network to guide the learning of a reliable latent label. We present a binary policy to determine whether latent label is from distant supervision or the prediction of extraction network, which is the core of our label denoising mechanism.
- evaluate the proposed method and discuss the results. The results demonstrate that our approach can outperform the state-of-the-art methods and RL is effective for latent label prediction.

The remainder of this paper is organized as follows. Section II provides the background knowledge. Section III introduces our proposed methods. Section IV reports and discusses the experimental results in detail. Section V summarizes the related work. Section VI concludes the whole

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2930340, IEEE Access

paper and gives the future work.

II. PRELIMINARY

A. RELATION EXTRACTION

First, we use some concepts and symbols to describe the relation extraction task more clearly. A KB contains the sets of entities $\mathcal{E}$, relation types $\mathcal{R}$ and relational facts $\mathcal{F}$. A fact $y_i(h_i, t_i) \in \mathcal{F}$ denotes that head entity $h_i \in \mathcal{E}$ and tail entity $t_i \in \mathcal{E}$ have a relation $y_i \in \mathcal{R}$. Through entity linking between KB and text corpus, we can create training data of relation extraction automatically. In multi-instance learning strategy, the training instances can be divided into multiple bags $\{B_1, B_2, ...\}$ with corresponding bag labels $\{y_1, y_2, ...\}$, and each bag B_i consists of multiple instances (sentences) $\{x_1^{(i)}, x_2^{(i)}, ...\}$ that mention entity pair h_i and t_i. Specially, if h_i and t_i are not related in KB, then we define their relation as NA, which is regarded as a negative class.

Following multi-instance learning, our goal of distantly supervised relation extraction task is to predict bag-level labels between entity pairs. In general, we will learn a mapping from bags to a probability distribution over relation classes $B \rightarrow p(y|B)$ as relation extractor. As prior methods of label denoising [13], [14], we will rectify the noisy labels from distant supervision with new labels during training. For the sake of simplicity, we refer to the distantly supervised label as *DS label* and the corrected new label as *latent label*.

B. REINFORCEMENT LEARNING

In our paper, the relation extraction task can be defined as a typical decision-making problem of RL for latent-label selection. Generally, in RL system, the learner and decision maker is called *agent*, which can be controlled by machine learning algorithms. The external thing that interacts with agent is called *environment*. The core idea of RL is to learn with the interaction between RL agent and environment. More specifically, RL agent observes a *state* s_t from its environment at time step t and takes an *action* a_t in state s_t. Based on the current state and the selected action, environment transitions to a new state s_{t+1}. Then environment provides a *reward* r_{t+1} to RL agent as feedback. With trial-and-error learning, RL agent can sense the state of its environment and take actions that affect the state in response to rewards obtained from the environment.

The reward can be a function of states and the selected actions, which is the primary basis for changing the behaviors of the RL agent. The broken subsequences of agent–environment interaction are called *episodes* and each episode ends in the *terminal state* at the final time step T. Like a round of the game, the beginning of the next episode is a reset of a starting state, which is independent of the end of the previous episode. A *policy* π is a decision-making rule and can determine the taken actions of RL agent at a given time. Generally speaking, it can be regarded as a mapping function $\pi(a|s)$ from states to actions. The goal of RL problem is to learn a policy to maximize the expected reward.

III. METHODS

In this section, we will present the proposed RL based label-noise reduction framework for distantly supervised relation extraction task. As illustrated in Figure 2, the framework mainly includes two modules – Extraction Network (ENet) and Policy Network (PNet). The ENet can predict the relational score of the input bag, from which we can represent the state of RL. The PNet can learn a latent label by taking action under a policy. With the obtained latent labels, the ENet can provide rewards to the PNet. To describe the entire framework clearly, the process can be modeled as a loop as below:

1) The bags are encoded to vector representation and input to ENet.
2) Based on the output in the ENet, we define the state of RL.
3) We take actions in these states depending on the policy in the PNet and learn the latent labels.
4) The ENet is updated with the obtained latent labels.
5) We receive a delayed reward from the ENet and give it to the PNet.

A. POLICY NETWORK

We will describe the entire RL process in detail. To reduce the label noise of distant supervision, we present a policy network (PNet) to gradually guide the correction of noisy labels and generate latent labels, which basically is an agent of RL. More concretely, we input N bags to encode their representations and obtain the predicted score with the ENet. Once the ENet has a predictive power, the incorrect prediction may come from noisy labels. In order to capture the discriminative features of incorrectly labeled instances, we only focus on T wrongly labeled bags in the PNet. Then we represent the state of RL and take action according to the policy in the PNet. The action are designed as a binary option between predicted label and DS label, by which we can produce the latent label. With the learned latent labels, we can update the ENet to obtain the reward. This reward is offered to the PNet to optimize the policy. The joint learning of two modules can reduce noisy labels of distant supervision process and improve the classification performance effectively.

Next, we will introduce five important components in RL: state, action, policy, reward and objective function as below:

State. As the input bags of our task are independent of each other, it is not appropriate to model each bag directly into a state because the environment of RL must satisfy Markov Decision Process (MDP) [16]. Hence, in addition to the information of current bag, the state s_t at t time step also includes the information of the bags whose DS labels have been corrected in early states. We use a continuous real-valued vector $\mathbf{s}_t$ to represent the state, which is defined as the concatenation of following vector representations: 1) Predicted score vector from the ENet of current bag, which is calculated by Eq. (6); 2) One-hot vector of DS labels of current bag; 3) Average predicted score vectors of the corrected DS labels in early states.

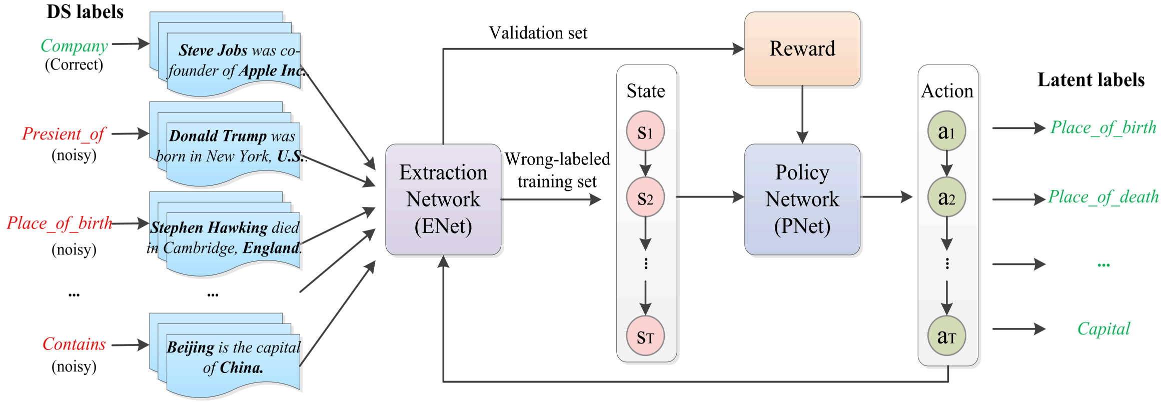


FIGURE 2: The proposed RL framework for label noise reduction.

Action. Since we intend to achieve noisy label corrections in our proposed denoising method, we just take into account the wrongly labeled bags where the predicted labels computed by Eq. (7) are different from DS label. Basically, a latent label is determined by the best trade-off between the DS label and model prediction. Therefore, we define a binary action a_t at t time step with two options {*DS label*, *Predicted label*} for a bag. Through taking the action in the current state, we can select DS label or predicted label as the latent label for current bag. Then the ENet can be trained with the learned latent label. Therefore, we will obtain a latent label to reduce the noisy label via selecting the action.

Policy. The policy based RL methods are expected to search for an optimal policy, from which we can sample the actions directly. Generally, the policy can be parameterized as a probability distribution function. To ensure exploration of RL, we sample the binary action by a stochastic policy $\pi(a_t|\mathbf{s}_t;\Theta) \in (0,1)$, which denotes the probability distribution of taking action a_t in state s_t given parameter Θ:

$$\pi(a_t|\mathbf{s}_t;\Theta) = \mathrm{F}(\mathbf{W}\mathbf{s}_t + \mathbf{b}) \tag{1}$$

where $\Theta = \{\mathbf{W}, \mathbf{b}\}$ is the parameter set of the PNet that denotes the training matrix and bias, and $\mathrm{F}(\cdot)$ indicates the sigmoid function.

According to the policy in the PNet, we are more likely to sample the action in a state with a higher probability during training. Due to the randomness of action sampling, we can achieve the exploration of latent-label learning in our task.

Reward. The target of RL is to maximize the expected reward under an optimal policy. Since our denoising method is to obtain latent labels that are generated by the taken actions, the design of reward is closely related to the learned latent labels. Intuitively, we expect to improve the relation classification performance of the ENet when the noisy labels are corrected by the PNet during training. Accordingly, similar to previous works [19], we use a performance-driven reward from the ENet to guide the learning of the PNet. Since interaction between PNet and ENet can break into episodes when all the T wrongly labeled bags are handled, where the terminal time step is T, thus the MDP is essentially episodic.

In particular, feedback from the ENet is only available when we complete the latent-label learning process in one episode. Thus, it is a typical delayed reward which can be obtained until the terminal state s_T is reached. In this situation, $r_t = 0 (t = 1, ..., T-1)$. More specifically, once all the actions are sampled, the latent labels are determined and regarded as the ground-truth labels to update the ENet. The classification results over validation set with the updated ENet can provide the reward signal to the PNet. We formulate the expected reward as below:

$$R = \frac{1}{|X^V|} \sum_{x_i \in X^V} \log p(y_i|x_i) \tag{2}$$

where x_i is a bag and y_i is the DS label of the bag. $p(y_i|x_i)$ is the predicted probability over DS label in updated ENet with latent labels, and X^V is the validation set. Clearly, we use an average likelihood over the number of the validation set $|X^V|$, which directly reflects the classification performance of the ENet.

Considering that ENet is supervised and updated by latent labels learned from the PNet, the taken actions of the PNet can affect the predictive power of the ENet. Therefore, the proposed reward can measure the effectiveness of the taken actions in the PNet.

Objective function. Empirically, we adopt policy gradient based methods [20] to optimize the PNet, which updates the policy based on the gradient of the objective function. For episode with T wrong-labeled training bags where the expected reward is R, we will maximize the objective function which is represented as:

$$J(\Theta) = \mathbb{E}_{\pi(a_{1:T}|s_{1:T};\Theta)} R \tag{3}$$

where Θ is the parameters of the PNet. More concretely, we use the REINFORCE rule [21] to iteratively update the parameters as below:

$$\nabla_\Theta J(\Theta) = \sum_{t=1}^{T} \nabla_\Theta \log \pi(a_t|\mathbf{s}_t;\Theta) R \tag{4}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2930340, IEEE Access

Naturally, based on the model optimization, we will receive a greater reward signal when the probability of the sampled actions for the correct latent-label is larger, where the update is effective for improving the classification performance of relation extractor.

B. EXTRACTION NETWORK

The target of extraction network (ENet) is to obtain the relational probability over class labels and provide information for state representation of RL. Built on the encoding process from plain text to vector representation, the output of ENet is semantically related to the relation pattern. The ENet consists of two encoders: 1) Sentence encoder, the embeddings of word and entity positions are brought together to learn a sentence representation with a neural network. 2) Bag encoder, the entity-pair information of multiple sentences in a bag is encoded into a representation according to an attention mechanism.

1) Sentence Encoder

Since the word embeddings [22], [23] have been proven effective for relation classification and extraction tasks [10], [24], we use the pre-trained embeddings to map a word into a contextual vector. Furthermore, we employ the entity position embeddings [24], [25] to locate the information of two entities in a sentence, which encodes the position features about the relative distances of each word to each entity into a real-valued vector. The semantic vector of a word in a sentence can be represented as a concatenation of the contextual vector and two position vector. Given the length of a sentence n, we can combine the n semantic vectors as the input sequence of sentence encoder.

For sentence encoding, we adopt the widely used architecture PCNN [10] to capture the sentence information. In order to get some information about the potential structure, according to two positions of the entity pair in a sentence, each feature map obtained by convolutional computation is divided into three pieces. Afterwards, we will use the max-pooling operation over each pieces and concatenate the three outputs as a sentence representation. For comparison, in addition to a standard CNN, we also use a Bidirectional Long Short Term Memory (BiLSTM) [26] with a max-pooling operation as the sentence representation.

2) Bag Encoder

With respect to an entity pair, we expect to assign higher weight for more informative sentences. Hence, a basic sentence attention mechanism [11] is used to obtain a weight α_i for the i-th sentence represent $\mathbf{e}_i$ in a bag. We can calculate a bag representation $\mathbf{x}$ as follows:

$$\mathbf{x} = \sum_i \alpha_i \mathbf{e}_i \qquad \alpha_i = \frac{\exp((\mathbf{e}_i \odot \mathbf{u})\mathbf{v})}{\sum_k \exp((\mathbf{e}_k \odot \mathbf{u})\mathbf{v})} \tag{5}$$

where $\mathbf{u}$ and $\mathbf{v}$ are two weighted vectors and $\odot$ is an element-wise multiplication operation. Based on the bag representation, we can compute the probability distribution over the relation classes with a softmax function as below:

$$\mathbf{p}_j = \frac{\exp(\boxed{\mathbf{M}\mathbf{x}_j + \mathbf{d}})}{\sum_k \exp(\mathbf{M}\mathbf{x}_k + \mathbf{d})} \tag{6}$$

where $\mathbf{p}_j$ denotes the predicted probability of the j-th relation, $\mathbf{M}$ and $\mathbf{d}$ are the parameter matrix and vector respectively. Let the number of labels be K, the predicted label can be represented as:

$$y^* = \arg\max\{\mathbf{p}_i, \mathbf{p}_2, ..., \mathbf{p}_K\} \tag{7}$$

In this paper, we employ the cross-entropy loss as the objective function of the ENet. Let $\hat{y}$ denote the latent label, we optimize the ENet by minimizing the following objective function:

$$\mathcal{L}(\hat{\mathbf{y}}, \mathbf{p}) = -\hat{\mathbf{y}} \log \mathbf{p} \tag{8}$$

where $\hat{\mathbf{y}}$ is the one-hot vector of the latent label. And we will use the mini-batch Adam optimizer [27] to update all the parameters of ENet.

C. MODEL TRAINING

In our approach, the pre-training strategy is of great importance. First, we pre-train the ENet with DS labels to get the predictive power of relation extractor. Afterwards, we pre-train the PNet as a binary classifier by regarding the soft labels [13] as the ground-truth while freezing pre-trained parameters of the ENet. Finally, we jointly train the ENet and PNet, the detailed joint training process is presented in Algorithm 1.

ALGORITHM 1: Joint training of PNet and ENet

Input: Training set X^T, validation set X^V, pre-trained ENet parameters Ψ_0 and PNet parameters Θ_0
Output: ENet parameters Ψ, PNet parameters Θ

1: **for** episode l=1 to L **do**
2:     Calculate the predicted score of X^T with Eq. (6)
3:     Obtain the wrongly labeled training set X^W and represent the state of RL
4:     **for** $x_t \in X^W$ **do**
5:         Sample the action a_t by policy $\pi(a_t|\mathbf{s}_t;\Theta)$ and obtain the latent label
6:     **end for**
7:     Update the parameter Ψ of ENet with latent labels
8:     Calculate the predicted score of X^V via Eq. (6)
9:     Obtain the expect reward with Eq. (2)
10:     Update the parameter Θ of PNet via Eq. (4)
11: **end for**

As described in lines 2-3, in order to represent the state of RL, ENet provides the predicted score over DS labels and obtains the wrongly labeled training set. Then PNet samples

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2930340, IEEE Access

the action according to Eq. (1) and obtains the latent label set in lines 4-5. The parameter set of the ENet is updated with the obtained latent labels in line 7. Afterwards, we compute the predicted score over the validation set and obtain the expected reward with Eq. (2) as lines 8-9. At last, we update the parameters of the PNet in line 10. Specially, following the work of Feng et al. [17], we also update the network more slowly in the joint training process. The parameter sets are formulated as $\Theta = \lambda\Theta + (1-\lambda)\hat{\Theta}$ and $\Psi = \lambda\Psi + (1-\lambda)\hat{\Psi}$, where λ is the weighted factor, $\hat{\Theta}$ and $\hat{\Psi}$ denote the parameters of the previous episode.

IV. EXPERIMENTS

In this section, we will evaluate the performance of the proposed method and compare it with the state-of-the-art distantly supervised relation extraction methods.

A. DATASET AND EVALUATION METRICS

We perform the experiment on a widely used benchmark dataset [7] for distantly supervised relation extraction task. The dataset is created by linking the entities in a human-curated KB Freebase [28] to the New York Times (NYT) corpus and consists of 52 positive relation classes and a negative (NA) class. The statistics of the used dataset are shown in Table 1.

TABLE 1: Statistics of dataset.

Type	Sentence	Entity Pairs	Relational Facts
Training set	522,611	281,270	18,252
Test set	172,448	96,678	1,950

As the common evaluation metrics of prior methods [10], [11], [13], we use two ways to evaluate our method: *held-out evaluation* and *manual evaluation*. The held-out evaluation compares the predicted labels of test set with the DS labels and reports the results with precision-recall curves, which gives an approximate measure that avoids human involvement. For the manual evaluation, we report the top N Precision (P@N) to eliminate the noisy labeling problem. Additionally, we manually check the accuracy of label correction and study the typical cases in detail.

B. EXPERIMENTAL SETUP

Our experiments are performed with Python 3.5 and Tensorflow 1.4.0 on a single Nvidia Titan X GPU. The original training set is split into a new training set and a validation set with a ratio of 2:1. Similar to previous works, we use the pre-trained 50-dimension word embeddings [11] and 5-dimension position embeddings. We set the window size and the number of feature maps for CNN and PCNN to 3 and 230, and the number of hidden units for BiLSTM to 230. Moreover, we use dropout [29] to alleviate the overfitting problem, whose dropout rate is set to 0.5. The learning rate and batch size for Adam optimizer [27] are set to 0.001 and 50 respectively. The pre-training parameters of the ENet follow the work of Sun et al. [15]. Especially for RL settings, we set the number of actions sampling during training to 3 at each episode, and the pre-training epoch of PNet to 3. The weighted factor λ in the joint training is set to 0.01. The entire joint training process ends with a maximum expected reward or up to 100 iteration epochs.

C. RESULTS AND DISCUSSION

We use seven baseline methods to show the effectiveness of our approach, including three traditional handcrafted feature based methods and four neural network based methods:

Mintz [6] is a traditional distantly supervised method based on the strong assumption that collects features from all sentences.

MultiR [8] and **MIMLRE** [30] are the traditional multi-instance learning and multi-instance multi-label methods respectively.

PCNN+ONE [10] employs PCNN to extract the sentence features, and just uses the most likely sentence in a bag for relation prediction.

PCNN [11] combines PCNN with a selective attention mechanism for sentence-level denoising.

PCNN+Soft-label [13] is a label-level denoising method based on the PCNN.

BGWA [31] achieves word-level denoising by using two word attention neural models.

PCNN+RL is the proposed model of our paper, whose pre-training of ENet module is based on the PCNN model. For fair comparison with PCNN-based models, we reproduce the results of PCNN and PCNN+Soft-label based on their settings in our experimental environment, the results of other compared methods are from their reports.

1) Held-out Evaluation

We compare the performance of different methods with the precision-recall curves of Figure 3. Clearly, we can find that: 1) The traditional feature engineering-based methods [6], [8], [30] perform much worse than the neural network-based methods [10], [11], [13], [31], which presents the effectiveness of using neural networks to extract semantic features in relation extraction task with noisy labels. 2) PCNN [11] has a significant improvement compared with PCNN+ONE [10], which proves that using sentence-level denosing within multiple sentences is more valid than just one sentence. 3) BGWA [31] performs better than PCNN model, which shows that word-level denoising is also effective for relation extraction. 4) Combining Soft-label [13] with PCNN can achieve higher precision than BGWA and PCNN, which indicates that label-level denoising can improve the relation extraction performance further. 5) Our proposed PCNN+RL model has the highest precision almost when the recall is greater than 0.05. Following the prior works [10], [13], when the recall is less than 0.05, we manually check the wrong predicted instances and find that most of them are caused by noisy labels in the test set, which is shown in Table 4 as follows.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2930340, IEEE Access

Accordingly, the result fully demonstrates the effectiveness and robustness of our RL method for label denoising.

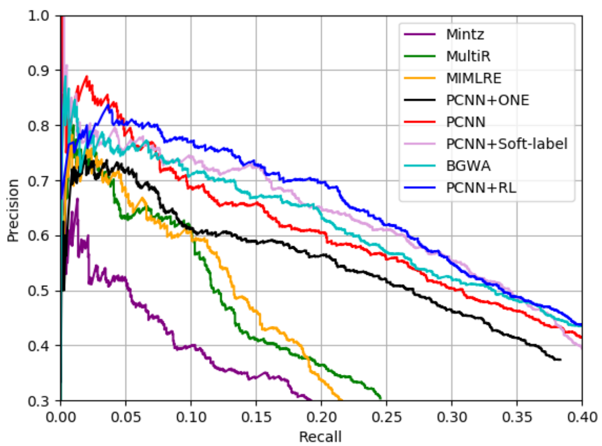


FIGURE 3: Performance comparison with previous methods.

To verify the generalization capability of our proposed RL methods for sentence representation, we replace the PCNN model with CNN and BiLSTM respectively in the sentence encoding layer of the ENet module. From Figure 4 we can observe that combining RL mechanism can remarkably improve the performance of CNN and BiLSTM baselines. With respect to CNN based model, using soft-label mechanism just performs better than the baseline when the recall is greater than 0.05 and less than 0.25. Nevertheless, utilizing RL mechanism has a significant improvement for the CNN baseline and the soft-label model when the recall is greater than 0.05. For the BiLSTM, we observe that soft-label method fail to improve BiLSTM baseline while our RL method can notably improve it. Besides, we show the results without the PNet pre-training for PCNN, CNN and BiLSTM. And we find that all the networks with the PNet pre-training perform better than without the pre-training obviously. Thus, our RL based label denoising method can be effectively generalized to different sentence encoders including CNN and BiLSTM. Moreover, the pre-training strategy is beneficial to our method.

Since the precision-recall curve fluctuations of various neural models are not obvious when the recall is greater than 0.05, for a more specific explanation, we report the top-ranked precision for different recalls 0.1/0.2/0.3/0.4 and their mean values in Table 2. And the best results are emphasized with bold formatting. For each neural model, combining RL can remarkably improve its baseline and perform much better than soft-label denoising method. Compared with the 8.6% and 8.5% improvement of CNN and BiLSTM baselines, undoubtedly, our proposed PCNN+RL achieves a better performance than the other two neural models and remains the best mean precision over these top 4 different recalls, which has a 10.8% improvement of PCNN baseline.

TABLE 2: Top-ranked precision of various neural models for different recalls.

Method Precision(%)	Recall 0.1	0.2	0.3	0.4	Mean
PCNN+ONE	61.3	56.4	46.5	-	-
PCNN	68.7	60.5	50.4	41.4	55.3
+Soft-label [13]	74.1	65.1	55.4	39.5	58.5
+RL	77.1	**69.5**	**55.0**	**43.7**	**61.3**
BGWA [31]	70.9	63.9	52.4	43.5	57.5
CNN	70.7	57.6	48.9	39.5	54.2
+Soft-label [13]	75.3	61.9	47.1	34.7	54.5
+RL	78.3	61.7	53.2	42.2	58.9
BiLSTM	75	59.0	47.9	37.1	54.8
+Soft-label [13]	60.7	45.4	30.8	7.9	36.2
+RL	**79.6**	64.8	52.5	41.2	59.5

As the prior works [11], [13], to verify that our RL method can benefit the relation extraction of entity pairs with multiple sentences, we report the top N precision (P@N) of the bags with more than one sentence in Table 3. More specifically, we randomly select one, two and all sentences from these bags in test data to predict the relation, and present the P@100, P@200, P@300 and their mean value for each neural model. We can see that combining soft-label mechanism can improve the results of PCNN and CNN baselines to some extent, but fail to improve the BiLSTM baseline effectively. Obviously, our RL method performs better than the soft-label method in all neural baselines. In particular, BiLSTM+RL almost achieves the best P@N results in all of these settings. These implications demonstrate that RL can improve the sentence-level denoising of the selective attention mechanism.

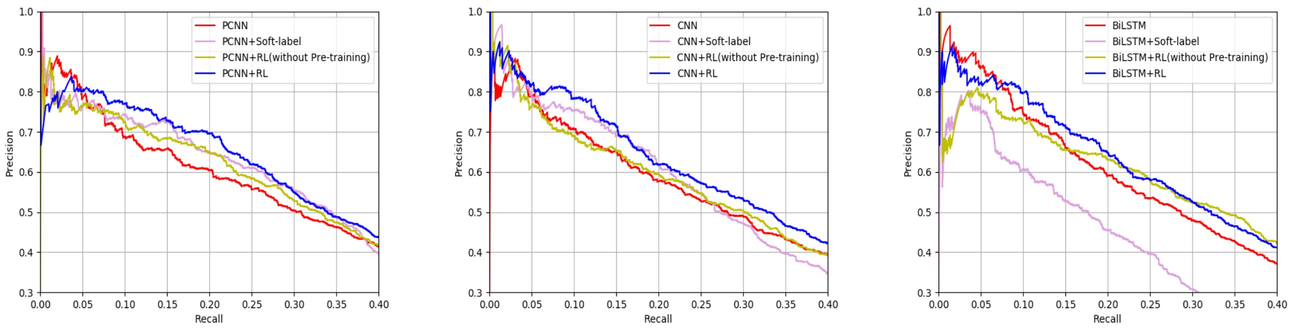
2) Manual Evaluation

As mentioned before, the held-out precision-recall curves drop dramatically at very low recall in Figure 3 due to the noisy label issue from distant supervision. Thus, following the work [10], we manually check the relational facts with high predicted probabilities and report the top 100/200/500 precision and their mean value in Table 4. We can observe that: 1) Almost for all the compared methods, the results from manual evaluation perform better than the results from held-out evaluation. 2) Our proposed PCNN+RL has the highest precision at different top N levels. On the whole, our method achieves the best performance for both held-out and manual evaluation.

3) Case Study

By manually examining 200 random bags, we find that our model can achieve a high correction accuracy 90.5%(181/200). Table 5 shows some typical cases of noisy-label corrections with our RL method. First, the noisy labels of some false positives can be corrected by our model. For example, despite there is a relation *Place_of_death* between *Kenneth Macmillan* and *London* in KB, the first instance does not explicitly indicate this relation. Similarly, the second sentence also fails to express the relation *Place_of_birth* between *Mark Aguirre* and *Chicago*. For these cases, our

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2930340, IEEE Access



(a) PCNN (b) CNN (c) BiLSTM

FIGURE 4: Performance comparison based on different sentence encoders. (a) PCNN. (b) CNN. (c) BiLSTM.

TABLE 3: Top N precision for relation extraction in the bags 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 [13]	76	72	68	72	80	76	69.7	75.2	81	77	71.7	76.6
+ RL	**81**	**75.5**	66.7	**74.4**	82	77.5	71	76.8	82	79	**76.3**	79.1
CNN	73	69	63.3	68.4	78	71	64.3	71.1	79	71.5	68	72.8
+ soft-label [13]	79	72	63	71.3	80	75.5	67	71.2	85	79.5	70	77.2
+RL	82	72	65.3	73.1	82	77	69.3	76.1	83	78.5	73	78.2
BiLSTM	80	68.5	60.7	69.7	80	70.5	64	71.5	83	74.5	67	74.8
+ soft-label [13]	68	52.5	46.3	55.6	74	62	55.7	63.9	79	67	58.7	68.2
+ RL	**81**	74	**67.7**	72.3	**86**	**80**	**75**	**80.3**	**86**	**81.5**	75.3	**80.9**

TABLE 4: Top N precision upon manual evaluation.

P@N(%)	**100**	**200**	**500**	**Mean**
Mintz [6]	77	71	55	67.7
MultiR [8]	83	74	59	72.0
MIMLRE [30]	85	75	61	73.7
PCNN+ONE [10]	86	80	69	78.3
PCNN [11]	86	82	71	79.7
PCNN+Soft-label [13]	87	84	74	81.7
PCNN+RL	**89**	**86**	**78**	**84.3**

method can effectively correct their noisy positive labels with a NA label. In particular, our approach does not correct the false negatives. We assume that the learning of our latent labels is related to class imbalance between positives and NA, which is consistent with the static presetting of higher confidence for NA class in soft-label method [13].

Besides, our proposed denoising method can reduce the noise of multi-label instances. For example, there are two relations – *Place_of_death* and *Nationality* between entity pair *Yehuda Amichai* and *Israel* in KB, our model can identify the right relation *Nationality* and correct the noisy label *Place_of_death* that is irrelevant to the instance. For another example, the relational pattern of the instance can be matched to *Contains(B, A)*, thus our model corrects the unspecific relation *Administrative_divisions* as the common one. As a result, our label denoising method can effectively correct the noisy labels via RL.

At last, we will discuss why the performance is improved in our model from a theoretical and experimental perspective. As proven in previous works [13]–[15], the label-level denoising methods can utilize the semantic information from correctly labeled instances to correct the noisy labels and improve the relation extraction performance effectively. The core of label denoising is the interaction between semantic information and distantly supervised labels. Based on this idea, our method uses RL to learn the interaction, which is theoretically feasible. Empirically, we find that the function design and pre-training strategy are critical to our model. In fact, we perform the experiments in different aspects, including the multiple choices of hyper-parameters, state representation, policy function and reward function, which are too trivial to be listed in our paper. And the components of our proposed framework are the optimal selection by multiple experiments. Therefore, our model can improve the performance of relation extraction theoretically and experimentally.

V. RELATED WORK

Distant supervision [6] was originally introduced to scale relation extraction to large-scale corpus, which can automatically label a large amount of training data by weakly supervised with external KBs. Through the alignment of triples in KBs with the sentences in unstructured text, the paradigm extracts features from all sentences and trains a classifier to predict the relation between two entities. However, the hypothesis that all sentences can indicate the relation in KB is too strong and always leads to wrong labeling challenge. Afterwards, multi-instance learning [7] is proposed to alleviate the problem, which assumes that at least one sentence

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2930340, IEEE Access

TABLE 5: Case study.

DS label	Latent label	Instances
Place_of_death	**NA**	*Kenneth Macmillan*'s "Manon", based on that hoary 18th-century French tale of a good girl gone from bad to worse, was created in 1974 at the Royal Ballet in *London*.
Place_of_birth	**NA**	Rebounds Larry Brown, the assistant *Mark Aguirre* and Isiah Thomas, the team president, flew to *Chicago* to attend the funeral of Ray Meyer, the former DePaul coach.
Place_of_death	**Nationality**	No reading of a poem by the Israeli poet *Yehuda Amichai* in honor of my late grandfather who once went to *Israel*.
Administrative_divisions	**Contains**	Another breakthrough for Poetica Musica came four years ago, when the state department sponsored its tour to Damascus, Homs and *Lattakia*, *Syria*.

that mentions the two entities can express the relation in KB. Subsequently, multi-instance multi-label [8], [30] is presented to handle the overlapping relations issue. Nevertheless, these early approaches require linguistic experts to design features elaborately, which is extremely limited by time and labour. Besides, handcrafted features are always generated by some NLP tools such as dependency parsing, which may easily propagate parsing errors and reduce the performance of downstream tasks [32].

In recent years, deep learning has been widely used as a promising approach for a variety of NLP tasks, which can automatically extract semantic features that fit a specific problem without relying on handcrafted features. In terms of relation extraction task, it has been proven that neural network based methods [10]–[12] achieve better performance than feature engineering based methods [7], [8], [30]. At first, Zeng at al. [10] adopt Piecewise Convolutional Neural Network (PCNN) to learn the sentence representation, and only select a valid sentence with the highest probability to train the relation extractor. Considering that most of information may be lost in the neglected sentences, sentence attention [11], [12] is proposed to learn higher weight for more effective sentences. Therefore, semantic information of multiple weighted sentences can be used to predict the relation more efficient. Besides, word attention [31] is used to determine the weight of words in a sentence. Qin et al. [33] employ generative adversarial network based method to provide a cleaned dataset for relation extraction task, which treat the positive samples from the generator as the negative samples to train the discriminator. And Vashishth et al. [34] utilize additional side information from KBs to predict relation, and exploit Graph Convolution Network (GCN) to encode the syntactic information from unstructured text.

RL [20] is a general way to learn optimal solutions by trial and error, and deep learning enables RL to extend to more complex decision-making settings such as high-dimensional state space [16]. In addition to sentence denoising methods above, RL is proposed as an effective method to reduce the impact of noisy sentences. Feng et al. [17] present a RL based model for sentence level relation extraction on the cleaned data. The model consists of an instance selector that chooses high-quality sentences with RL and a relation classifier which provides delayed rewards for the quality of the selected sentences to the instance selector. Zeng et al. [18] employ RL to learn sentence relation extractor, and follow expressed-at-least-once assumption to determine the long term reward. Then Qin et al. [19] adopt RL to learn a sentence-level false-positive indicator, which aims to identify false positives for each relation. Recognized false positives are redistributed into the negative set to train the relation classifier, and a result-driven reward is used to model the difference between the adjacent epochs. These mentioned RL methods focus on sentence denoising and treat the labels from distant supervision as the ground-truth.

Another solution for noisy labeling problem in distantly supervised relation extraction is label denoising [13]–[15]. Liu et al. [13] propose to correct the wrong labels with new soft labels during training by utilizing the semantic information from true positives. To obtain the soft label, a joint score function that combines the predicted score from the neural network and static confidence of distantly supervised labels is designed, which is a simple but efficient noise-tolerant method. Sun et al. [14] present three impact factors of label denoising for distantly supervised relation extraction including prior knowledge, labeling assumption and confidence level, which can affect word-level, sentence-level and label-level denoising correspondingly. Their experimental results show that partial confidence for distantly supervised labels can significantly improve the denoising performance. Then a multi-head self-attention network based confidence curriculum learning method [15] is proposed to address the label denoising problem.

More recently, a deep pattern diagnosis framework [35] is proposed for distantly supervised relation extraction, which uses RL to extract relation-specific patterns. After fusing DS labels and refined patterns, the method helps diagnose the labeling noises and improve the relation extraction performance. And a hierarchical RL model [36] is proposed to address relation mention extraction, which includes a high-level sentence selector to remove noisy sentences and a low-level mention extractor. Besides, Zhu et al. [37] apply graph neural network with generated parameters to process multi-hop relational reasoning. And Christopoulou at al. [38] propose a graph walk based neural network that considers interactions among multiple entity pairs in relation extraction. As for inter-sentence relation extraction, a GCN model

[39] on a document-level graph is used to capture local and non-local dependencies. In terms of joint entity and relation extraction task [40], a hierarchical RL framework [41] is proposed to decompose the task into a high-level relation detection and low-level entity recognition, which can enhance the interaction between two subtasks and extract overlapping relations more effectively. These mentioned methods above inspire us to explore RL based label denoising strategy.

VI. CONCLUSIONS

In this paper, we propose a RL based method to correct the noisy labels in distantly supervised relation extraction task. In our model, the ENet obtains relational score for state representation of RL, then the PNet generates a latent label by taking action based on the designed policy to update the ENet, finally the optimized ENet provides rewards to update the PNet. Joint learning of two modules can rectify the noisy labels generated by distant supervision and improve the relation extraction performance. Experimentally, our method can remarkably outperform the state-of-the-art distantly supervised relation extraction methods. Besides, our RL algorithm can be effectively generalized to different sentence encoder including CNN and BiLSTM. Case studies demonstrate that our approach can reliably identify the latent labels.

In the future, we plan to explore the generative adversarial network based label-level denoising method. In addition, we will take into consideration using external knowledge sources to supervise the corrections of noisy labels for each sentence.

REFERENCES

[1] F. Wu and D. S. Weld, "Open information extraction using wikipedia," in Proceedings of the 48th annual meeting of the association for computational linguistics. Association for Computational Linguistics, 2010, pp. 118–127.

[2] X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, and W. Zhang, "Knowledge vault: A web-scale approach to probabilistic knowledge fusion," in Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2014, pp. 601–610.

[3] Y. Luan, L. He, M. Ostendorf, and H. Hajishirzi, "Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction," 2018, pp. 3219–3232.

[4] A. Fader, L. Zettlemoyer, and O. Etzioni, "Open question answering over curated and extracted knowledge bases," in Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2014, pp. 1156–1165.

[5] K. Xu, S. Reddy, Y. Feng, S. Huang, and D. Zhao, "Question answering on freebase via relation extraction and textual evidence," in Proceedings of ACL, 2016, pp. 2326–2336.

[6] 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. Association for Computational Linguistics, 2009, pp. 1003–1011.

[7] S. Riedel, L. Yao, and A. McCallum, "Modeling relations and their mentions without labeled text," Machine learning and knowledge discovery in databases, pp. 148–163, 2010.

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

[9] A. Ritter, L. Zettlemoyer, O. Etzioni et al., "Modeling missing data in distant supervision for information extraction," Transactions of the Association for Computational Linguistics, vol. 1, pp. 367–378, 2013.

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

[11] Y. Lin, S. Shen, Z. Liu, H. Luan, and M. Sun, "Neural relation extraction with selective attention over instances." in ACL, 2016.

[12] G. Ji, K. Liu, S. He, and J. Zhao, "Distant supervision for relation extraction with sentence-level attention and entity descriptions." in AAAI, 2017, pp. 3060–3066.

[13] T. Liu, K. Wang, B. Chang, and Z. Sui, "A soft-label method for noise-tolerant distantly supervised relation extraction," in Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 1791–1796.

[14] T. Sun, C. Zhang, and Y. Ji, "Factors impacting the label denoising of neural relation extraction," in Proceedings of the 12th International Conference on Algorithmic Aspects in Information and Management (AAIM), 2018, pp. 12–23.

[15] 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. 54 472–54 482, 2019.

[16] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "A brief survey of deep reinforcement learning," arXiv preprint arXiv:1708.05866, 2017.

[17] J. Feng, M. Huang, L. Zhao, Y. Yang, and X. Zhu, "Reinforcement learning for relation classification from noisy data," in Proceedings of AAAI, 2018.

[18] X. Zeng, S. He, K. Liu, and J. Zhao, "Large scaled relation extraction with reinforcement learning," in AAAI, vol. 2, 2018, p. 3.

[19] P. Qin, W. Xu, and W. Y. Wang, "Robust distant supervision relation extraction via deep reinforcement learning," arXiv preprint arXiv:1805.09927, 2018.

[20] R. S. Sutton and A. G. Barto, Reinforcement learning: An introduction. MIT press, 1998.

[21] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," Machine learning, vol. 8, no. 3-4, pp. 229–256, 1992.

[22] T. Mikolov, W.-t. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations." in HLT-NAACL, vol. 13, 2013, pp. 746–751.

[23] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation." in EMNLP, vol. 14, 2014, pp. 1532–43.

[24] D. Zeng, K. Liu, S. Lai, G. Zhou, J. Zhao et al., "Relation classification via convolutional deep neural network." in COLING, 2014, pp. 2335–2344.

[25] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," Journal of Machine Learning Research, vol. 12, no. Aug, pp. 2493–2537, 2011.

[26] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in Acoustics, speech and signal processing (icassp), 2013 ieee international conference on. IEEE, 2013, pp. 6645–6649.

[27] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.

[28] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: a collaboratively created graph database for structuring human knowledge," in Proceedings of the 2008 ACM SIGMOD international conference on Management of data, 2008, pp. 1247–1250.

[29] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting." Journal of machine learning research, vol. 15, no. 1, pp. 1929–1958, 2014.

[30] 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. Association for Computational Linguistics, 2012, pp. 455–465.

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

[32] R. McDonald and J. Nivre, "Characterizing the errors of data-driven dependency parsing models," in Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), 2007.

[33] P. Qin, W. Xu, and W. Y. Wang, "Dsgan: Generative adversarial training for distant supervision relation extraction," arXiv preprint arXiv:1805.09929, 2018.
[34] S. Vashishth, R. Joshi, S. S. Prayaga, C. Bhattacharyya, and P. Talukdar, "Reside: Improving distantly-supervised neural relation extraction using side information," in EMNLP, 2018, pp. 1257–1266.
[35] S. Zheng, P. Yu, L. Chen, L. Huang, and W. Xu, "Diag-nre: A deep pattern diagnosis framework for distant supervision neural relation extraction," in ACL, 2019.
[36] J. Feng, M. Huang, Y. Zhang, Y. Yang, and X. Zhu, "Relation mention extraction from noisy data with hierarchical reinforcement learning," in AAAI, 2019.
[37] H. Zhu, Y. Lin, Z. Liu, J. Fu, T.-s. Chua, and M. Sun, "Graph neural networks with generated parameters for relation extraction," arXiv preprint arXiv:1902.00756, 2019.
[38] F. Christopoulou, M. Miwa, and S. Ananiadou, "A walk-based model on entity graphs for relation extraction," arXiv preprint arXiv:1902.07023, 2019.
[39] S. K. Sahu, F. Christopoulou, M. Miwa, and S. Ananiadou, "Inter-sentence relation extraction with document-level graph convolutional neural network," arXiv preprint arXiv:1906.04684, 2019.
[40] R. J. Kate and R. J. Mooney, "Joint entity and relation extraction using card-pyramid parsing," in Proceedings of the Fourteenth Conference on Computational Natural Language Learning. Association for Computational Linguistics, 2010, pp. 203–212.
[41] R. Takanobu, T. Zhang, J. Liu, and M. Huang, "A hierarchical framework for relation extraction with reinforcement learning," in AAAI, 2019.

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. degree, the M.S. degree 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 interests include deep learning, ubiquitous computing, and data mining. Her research has been supported by the Ministry of Science and Technology of China and the National Natural Science Foundation of China.

YANG JI received the Ph.D. degree in electronic engineering from the Beijing University of Posts and Telecommunications (BUPT) in 2002. He is currently a Professor with BUPT. 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 system, 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. He is currently with the State Key Laboratory of Networking and Switching Technology, BUPT, where he is also with the Institute of Sensing Technology and Business. He is involved in ubiquitous networking and service computing. His current interests lie in the user behavior modeling and analysis in mobile Internet and social networks.