The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Large Scaled Relation Extraction with Reinforcement Learning

Xiangrong Zeng,[12] **Shizhu He,**[2] **Kang Liu,**[12] **Jun Zhao**[12]
[1]University of Chinese Academy of Sciences, Beijing, 100049, China
[2]National Laboratory of Pattern Recognition (NLPR), Institute of Automation
Chinese Academy of Sciences, Beijing, 100190, China
{xiangrong.zeng, shizhu.he, kliu, jzhao}@nlpr.ia.ac.cn

Abstract

Sentence relation extraction aims to extract relational facts from sentences, which is an important task in natural language processing field. Previous models rely on the manually labeled supervised dataset. However, the human annotation is costly and limits to the number of relation and data size, which is difficult to scale to large domains. In order to conduct largely scaled relation extraction, we utilize an existing knowledge base to heuristically align with texts, which not rely on human annotation and easy to scale. However, using distant supervised data for relation extraction is facing a new challenge: sentences in the distant supervised dataset are not directly labeled and not all sentences that mentioned an entity pair can represent the relation between them. To solve this problem, we propose a novel model with reinforcement learning. The relation of the entity pair is used as distant supervision and guide the training of relation extractor with the help of reinforcement learning method. We conduct two types of experiments on a publicly released dataset. Experiment results demonstrate the effectiveness of the proposed method compared with baseline models, which achieves 13.36% improvement.

Introduction

Relation Extraction (**RE**) devotes to extracting relational facts from sentences, which can be applied to many natural language processing (**NLP**) applications such as information extraction (Wu and Weld 2010) and question answering (Yih et al. 2015; Dai, Li, and Xu 2016). Given a sentence with an entity pair e_1 and e_2, this task aims to identify the relation between e_1 and e_2. For example, RE devotes to extracting the relation of *Steve Jobs* and *Apple* by given the first sentence in Figure 1.

RE has drawn much attention of many researchers in NLP field. (Zeng et al. 2014) is among the first work to apply neural networks in this task. They adopted Convolutional Neural Network (CNN) to automatically extract the sentence representations with the raw input words for RE, which achieved significant improvements compared with traditional models. (Zhou et al. 2016) applied attention mechanism with Long Short-Term Memory (LSTM) Networks to capture the semantic information in a sentence, which didn't utilize any features derived from lexical resources or NLP systems. They achieved higher F1 score than most baselines methods. These models are heavily relied on supervised dataset like ACE-05 (Grishman, Westbrook, and Meyers 2005) and SemEval-2010 task 8 (Hendrickx et al. 2009). However, supervised data relies on human annotation which is very costly. Therefore, generating data with human annotation is difficult to apply in the large scaled domain. How to extract relations in large domain dataset remains a challenging task.

To obtain large scaled relation extraction dataset, (Mintz et al. 2009) proposed Distant Supervision paradigm to automatically scale RE to large domains. They used the relational facts from large scaled Knowledge Bases (**KB**) to automatically align with texts. Specifically, for a triplet fact $r(e_1, e_2)$ in a KB, all sentences that mention both entities

Sentence	Relation
1. **Steve Jobs** and Wozniak co-founded **Apple** in 1976.	*Founder*
2. **Michael Jordan** is an American retired professional **basketball player**.	*Career*
3. **Washington D.C.** is the capital of **United states**.	*CapitalOf*
......	......

Figure 1: Examples of relation extraction task. This task devotes to extract relations between an entity pair.

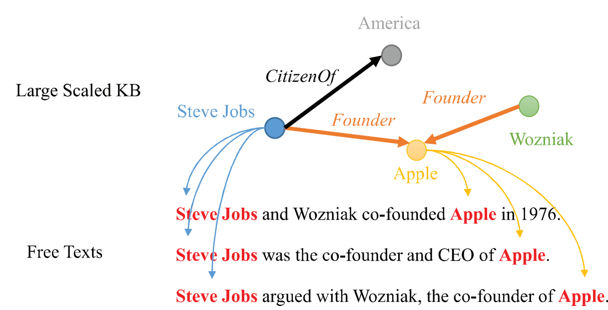

Figure 2: Align relational facts from KB with free texts to automatically generate large scaled dataset. However, sentences are not directly labeled and some of them (i.e. the third sentence in the figure) can not represent the relation of the entity pair.

e_1 and e_2 are aligned with relation r. We call the set containing an entity pair with sentences mentioned them as a **Bag**. Figure 2 shows this process. Although this method can obtain large scaled data and is easy to expand, sentences in distant supervised data are not directly labeled and not all sentences that mention an entity pair can represent the relation between them. In the example in Figure 2, there are no direct label for each sentences and the third sentence is not represent the *Founder* relation between *Steve Jobs* and *Apple*. In this paper, relations like *Founder* are called real relations. We also introduce NA relation, which represents no relation between two entities.

To learn relation extractor with large scaled distant supervised data, we draw insights from reinforcement learning (**RL**) methods.

The relation extractor is regarded as RL Agent and the goal is to achieve higher long-term reward. The agent reads the bag's sentences and outputs their extracted relations one by one. We integrate the predicted relations of sentences to predict the relation of the bag, which will be compared with the gold bag relation to determine the long-term reward. We then utilize it to train the relation extractor. Figure 3 shows this process.

In our model, we need to integrate the predicted sentence relations into bag relation so that we can compare it with the gold bag relation to determine the long term reward. We follow the expressed-at-least-once assumption (Riedel, Yao, and McCallum 2010) to predict the bag relation but rephrase it from the prediction aspect of view: *When predicting the relation of a bag, the bag is NA relation when and only when all sentences in bag represents NA relation, otherwise, the bag is the real relation represent by its sentences*.

Two types of experiments on a publicly released dataset demonstrate that our method outperforms the comparative baselines significantly, which achieves 13.36% improvement. We summarize our contributions as follows:

1. We apply reinforcement learning method to learn sentence relation extractor with the distant supervised dataset. The bag relation is used as distant supervision, which monitors the training of relation extractor.
2. Benefit from relation extractor, relations of bag's sentences are extracted, which is helpful to predict the bag relation.
3. We conduct two types experiments on two versions of a widely used dataset and outperform the comparative baselines significantly.

Related Work

Supervised Relation Extraction

Supervised relation extraction task aims to extract relations from sentences with supervised data. (Zeng et al. 2014) is among the first to apply neural networks to this task. They utilized the convolutional neural network to automatically extract features which not depend on traditional NLP tools and avoid the error propagation problem. (Xu et al. 2015) adopted LSTM networks along the shortest dependency path. (Zhou et al. 2016) proposed a novel model which

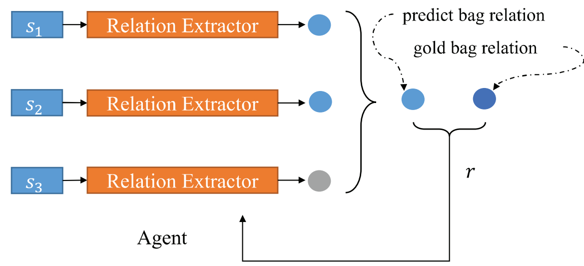

Figure 3: The process of reinforcement learning. Relation extractor is RL agent and r is the generated reward.

utilizes attention mechanism with Bidirectional LSTM networks and achieve promising results. But all of these models are built based on the supervised dataset.

Our model is different from them since we devote to learning relation extractor on the distant supervised dataset.

Distant Supervised Relation Extraction

Distant supervised relation extraction aims to extract the relation of the bag. Many researchers have focused on this task.

(Zeng et al. 2015) proposed PCNN to automatically extract features from sentences and applied MIL to select the most important sentence. Both (Lin et al. 2016) and (Ji et al. 2017) applied attention mechanism to alleviate the inference of noises. (Ji et al. 2017) also utilize the entity description as external information to improve performance. (Jiang et al. 2016) used cross-sentence max-pooling to take all sentences into consideration and considered the multi-label problem. (Lin, Liu, and Sun 2017) used cross-lingual attention to consider the information consistency and complementarity among cross-lingual texts. (Luo et al. 2017) used a dynamic transition matrix to characterize the noise and apply curriculum learning framework to guide training.

However, all of these models focus on bag relation extraction. Our model is different from them since we learn a sentence relation extractor.

Reinforcement Learning

We also relate to prior works on reinforcement learning. Reinforcement learning has been successfully applied to many games such as Go (Silver et al. 2016) and Atari games (Mnih et al. 2015). We get inspiration from the Atari Pong game with reinforcement learning. We also get inspired from the researchers that applied RL methods to NLP tasks. (Narasimhan, Kulkarni, and Barzilay 2015) used reinforcement learning for text-based games. (Narasimhan, Yala, and Barzilay 2016) applied reinforcement learning on information extraction task by acquiring external evidence and they achieved huge improvements compared with traditional extractors. (Li et al. 2016) used reinforcement learning for dialogue generation and fostered a more sustained dialogue, which manages to produce more interactive responses than standard methods. To apply Generative Adversarial Net (GAN) to generating sequences, (Yu et al.

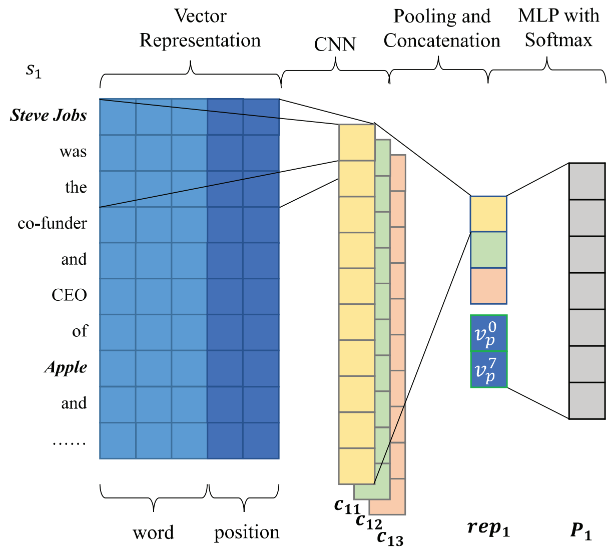

Figure 4: The structure of our relation extractor. Input a sentence s_i, the extractor output the probability distribution $\boldsymbol{P}_i$. We use $\boldsymbol{c}_{ij}$ to denote j-th feature map and $\boldsymbol{rep_i}$ denote the representation of s_i.

2017) considered the sequence generation procedure as a sequential decision-making process and treated the generative model as an agent of reinforcement learning. Their model significantly outperforms the baseline methods.

Our Model

To take the advantage of distant supervision, we treat the process of extracting the relations of every sentence in a bag as a reinforcement learning episode. In other words, given a bag, we first extract relation of every sentence independently. We then predict the bag relation based on the extracted relations and compare it with the gold bag relation. Finally, we use the result of the comparison to guide the training of relation extractor.

Relation Extractor

As shown in Figure 4, we use neural networks to build the relation extractor. Prior works like (Zeng et al. 2015; Lin et al. 2016; Ji et al. 2017) used PCNN to represent sentences. In this paper, we use a more straightforward way to model sentences which can achieve comparable performance with PCNN, but is easier for implementation and more efficient for calculation. Specifically, given a raw sentence s_i, we first split the sentence into tokens. Then we turn every token into dense vectors, which will be used as the inputs of the convolutional neural networks. Instead of using the piece-wise max-pooling, we directly use the normal max-pooling but concatenate the max-pooling result with the position embedding of two entities. We denote it as Position Enhanced (**PE**) CNN. Finally, a multi-layer perceptron with softmax is applied to output the probability of every relation (including NA).

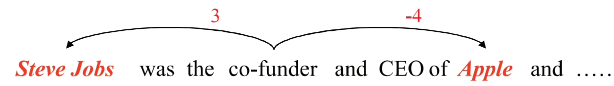

Figure 5: Relative positions of token *co-funder*. *Steve Jobs* and *Apple* are the entities.

Word Embeddings Word embeddings are low-dimensional vector of tokens, which are learned from the large unlabeled text. It has been used in many NLP tasks and shows its power. Every token is corresponding to a word embedding $\boldsymbol{v_w}$. In this paper, we train word embeddings on the New York Times (NYT) corpus with word2vec toolkit (Mikolov et al. 2013a; 2013b; Mikolov, Yih, and Zweig 2013).

Position Embeddings Position embeddings are low-dimensional vectors of positions. It has been successfully used in many works of relation extraction (Zeng et al. 2014; 2015; Lin et al. 2016; Ji et al. 2017). The relative position is the distance between token and entity. As shown in Figure 5 , the relative position from token *co-funder* to entity *Steve Jobs* and *Apple* is 3 and -4, respectively. Every relative position is corresponding to a dense vector $\boldsymbol{v_p}$ which is called position embedding.

The vector representation $\boldsymbol{v}$ is concatenated by word embedding $\boldsymbol{v_w}$ and position embedding $\boldsymbol{v_p}$ as shown in the *Vector Representation* part in Figure 4. We have $d_v = d_w + 2d_p$, where d_v, d_w and d_p are the dimension of $\boldsymbol{v}$, $\boldsymbol{v}_w$ and $\boldsymbol{v}_p$ respectively.

Convolution The convolution of $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ and $\boldsymbol{B} \in \mathbb{R}^{m\times n}$ is defined as

$$\boldsymbol{A} \otimes \boldsymbol{B} = \sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij} b_{ij} \qquad (1)$$

Let $\boldsymbol{v_j^i}$ denotes the vector of j-th token in sentence s_i and let $\boldsymbol{S_i}$ represents the matrix concatenated by $[\boldsymbol{v}_1^i; \boldsymbol{v}_2^i; ...; \boldsymbol{v}_{|s_i|}^i]$, where $|s_i|$ is the number of tokens contained by s_i. Given $\boldsymbol{S}_i$, we use a filter $\boldsymbol{W}_q \in \mathbb{R}^{w\times d_v}$ to extract local features from s_i. By sliding $\boldsymbol{W}_q$ along the sentence s_i, we could obtain a vector $\boldsymbol{c}_{iq} \in \mathbb{R}^{|s_i|-w+1}$ called feature map by following equation:

$$c_j^{iq} = f([\boldsymbol{v}_j^i; \boldsymbol{v}_{j+1}^i; ...; \boldsymbol{v}_{j+w-1}^i] \otimes \boldsymbol{W}_q + b) \qquad (2)$$

where $b \in \mathbb{R}$ is a bias and $f(\cdot)$ is non-linear function such as Tanh and ReLU. As shown in Figure 4, for sentence s_i, we apply three filters $\boldsymbol{W}_1, \boldsymbol{W}_2, \boldsymbol{W}_3$ and get three feature maps $\boldsymbol{c}_{i1}, \boldsymbol{c}_{i2}$ and $\boldsymbol{c}_{i3}$.

Pooling and Concatenation The max-pooling of vector $\boldsymbol{c}_{iq} = [c_1^{iq}, c_2^{iq}, ..., c_n^{iq}]$ is defined as $\max(c_1^{iq}, c_2^{iq}, ..., c_n^{iq})$. We then concatenate the pooling results with the positions embedding of entities. In the example in Figure 4, the position of entities are 0 and 7. We concatenate their position embeddings v_p^0 and v_p^7 with the maxpooling result to form the representation of sentence rep_1.

Multilayer Perception with Softmax After we get the representation $\boldsymbol{rep}_i$ of sentence s_i, we apply a single layer MLP to output the confidence vector $\boldsymbol{O}_i$. Then the conditional probability of j-th relation is

$$p(rel_j|\theta, s_i) = \frac{exp(o_j)}{\sum_{k=1}^{M} exp(o_k)} \tag{3}$$

Training with Reinforcement Learning

To learn the relation extractor without the direct guide, we introduce the policy gradient method in reinforcement learning.

Predict the Bag Relation To use the information of distant supervision (the bag relation), we need to predict the bag relation based on the relation of its sentences. In this paper, we follow the expressed-at-least-once assumption (Riedel, Yao, and McCallum 2010) to predict the bag relation, which we detailed as following rules:

- For sentence s_i, we select the relation with the highest probability as the extracted relation of s_i. That is, we extract relation rel_{j^*} from s_i, where $j^* = \arg\max p(rel_j|\theta, s_i)$.
- If all sentences in bag are extracted as NA relation, we predict the bag with relation NA;
- Otherwise, we ignore all sentences that predict NA relation. From the rest sentences, the extracted relation with the highest probability is regarded as the relation of bag[1].

In the episode of a bag, sentences are regarded as RL states and relations are regarded as RL actions. Our relation extractor is treated as RL agent. Suppose there are n sentences in a bag. Before the episode is finished (that is, not all sentences in the bag have been extracted), the rewards of the state are set to 0 (that is $r_i = 0, i = 1, ..., n-1$) since we don't know if this episode is good or not. Once the episode is finished, we predict the relation of the bag and the reward of the last state r_n is determined by the predicted results of the bag. If the predicted relation is the same as the gold one, we assign a positive value to r_n, otherwise, we assign a negative value. The advantage of state s_i is calculated by

$$R(s_i) = \sum_{j=i}^{n} \gamma^{n-j} r_j \tag{4}$$

where $\gamma \in (0, 1]$ is the discount factor. Since $r_i = 0, i = 1, ..., n-1$, $R(s_i)$ can be simplified as

$$R(s_i) = \gamma^{n-i} r_n \tag{5}$$

In this task, the order of sentences in bag should not influence the predicted result, so we set $\gamma = 1$ and we have

$$R(s_i) = r_n \tag{6}$$

In the experiment, r_n is set to $+1$ or -1.

[1]We only ignore NA sentences when predicting the bag relation, all sentences (including the NA sentences) are used when training, which is different from (Zeng et al. 2015)

For the example in figure 2, we first extract relations from s_1, s_2 and s_3 independently. Suppose the first sentence is predicted to relation *Founder* with probability 0.7, the second sentence is predicted to relation *Founder* with probability 0.8 and the third sentence is predicted to relation NA with probability 0.9. To predict the bag relation, we ignore the sentence that extracts NA relation regardless its probability. In this example, s_3 is ignored. Therefore, The bag is predicted to relation *Founder* because s_2 is extracted with the highest probability (0.8) and the extracted relation is *Founder*. Because the gold relation of this bag is *Founder* which is the same as the predicted relation, the episode reward will be set to +1.

Optimization We use REINFORCE (Williams 1992) algorithm to optimize the policy (sentence-level classifier) of our model. We use θ to represent the parameters of the entire model and all parameters will be trained together.

For the episode of a Bag with n sentences and the advantage of i-th sentence is $R(s_i)$, the objective function can be defined as:

$$J(\theta) = \mathbb{E}_{s_1,...,s_n} R(s_i) \tag{7}$$

where a^i is the predicted relation of s_i.

We can update the gradients of θ by using the likelihood trick (Williams 1992) as:

$$\nabla J(\theta) = \sum_{i=1}^{n} \nabla p(a^i|s_i, \theta) R(s_i) \tag{8}$$

where a^i is the action (relation) taken in state s_i.

To reduce variance and make training faster and more stable, we introduce a baseline b (Williams 1992). For a batch of data with N bags, the baseline is calculated as the mean of all advantages in batch:

$$b = \frac{\sum_{i=1}^{N} \sum_{j=1}^{n_i} R(s_j)}{\sum_{i=1}^{N} n_i} \tag{9}$$

Then we can update the gradients of θ by

$$\nabla J(\theta) = \sum_{i=1}^{N} \sum_{j=1}^{n_i} \nabla p(a^j|s_j, \theta)(R(s_j) - b) \tag{10}$$

Experiment

To evaluate the effectiveness of our extractor in large scaled data, we train and test our model in the distant supervised dataset. However, sentences in a distant supervised dataset are not directly labeled, which make the evaluation difficult. Therefore, we implement two type of experiments to demonstrate the effects of our proposed model.

We first manually label a test dataset and evaluate our sentence relation extractor directly. However, this evaluation method requires costly human annotation and can only be used in small-scale datasets.

To evaluate the model in large-scale test dataset, we test our model in distant supervised relation extraction task. This task focuses on the prediction of bag relation, therefore, the distant supervised data can be directly used for evaluation

Dataset		# Bags	# positive Bags	# sentences	# relations
SMALL	Train	65726	4266	112941	26
	Test	93574	1732	152416	26
LARGE	Train	281270	18252	522611	53
	Test	96678	1950	172448	53

Table 1: Dataset statistics. "# Bags" means the number of Bags.

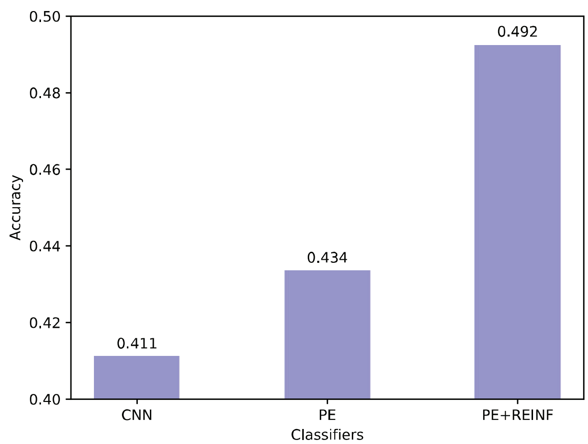

Figure 6: The accuracy of relation extractor based on CNN, PE and PE+REINF. It's obvious that our model leans a better relation extractor.

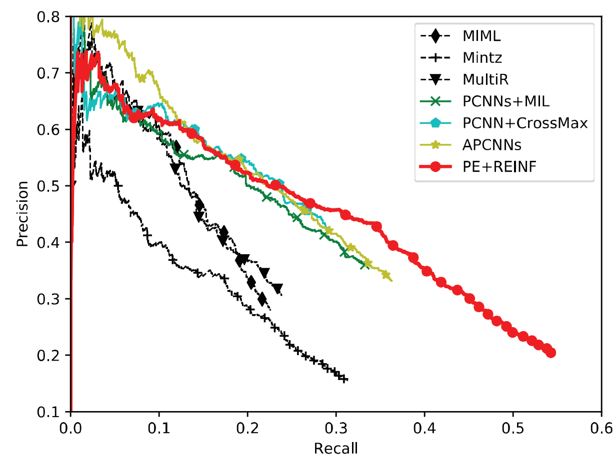

Figure 7: Results of different models on SMALL dataset.

without requiring human annotation. Although our model is not naturally designed for distant supervised relation extraction task, we can easily apply our model to it. The prediction of the bag heavily relies on the relation extractor, therefore, evaluation results in distant supervised relation extraction task can demonstrate the effectiveness of our model.

Dataset

In this paper, we use the widely used dataset developed by (Riedel, Yao, and McCallum 2010)[2] to evaluate our model. This dataset is created by aligning Freebase with New York Times (NYT) corpus. From which, 2005-2006 NYT corpus is used for training and 2007 corpus for testing. There are two versions of the dataset. The first version is comparably smaller, we denote the first version dataset as SMALL while the second as LARGE.

The SMALL dataset has two versions too. The original version of SMALL dataset is used by (Riedel, Yao, and McCallum 2010), (Hoffmann et al. 2011) and (Surdeanu et al. 2012). The filtered version of SMALL dataset is used by (Zeng et al. 2015), (Jiang et al. 2016) and (Ji et al. 2017). (Zeng et al. 2015) filtered the origin dataset by removing a) duplicated sentences in each bag; b) sentences which have more than 40 tokens between two entities; c) sentences with entity names that are substrings of other entity names in Freebase. In this paper, we use the filtered version of SMALL dataset[3].

The LARGE dataset is used and released by (Lin et al. 2016)[4]. It has a much larger training data size than SMALL.

We represent the detailed statistics of SMALL and LARGE dataset in Table1. The Bag with entity pair that indicates none NA relation is called positive Bag. For each dataset, we randomly select 20% training data as validation for tuning the parameters of our model and use the rest to train the models.

Implementation Detail

In our experiments, we follow the settings of (Zeng et al. 2015). The dimension of word embedding $|v_w|$ is set to 50, the dimension of position embedding $|v_p|$ is set to 5, the window size w of filters is 3 and the number of filters K is set to 230. The batch size is fixed to 50 and dropout probability fix to 0.5. When training, we use Adam (Kingma and Ba 2015) to optimize parameters. It's worth to mention that pretraining is important in reinforcement learning. The learning rate is reset to 0.0001 and batch size is set to 2000 when applying reinforcement learning.

Relation Extraction

We conduct the experiment of comparing different extractors on SMALL dataset. Since most of the entity pairs are negative (contains NA relation), if we randomly select sentences from the test set, models could easily achieve high

[2]http://iesl.cs.umass.edu/riedel/ecml/

[3]http://www.nlpr.ia.ac.cn/cip/~liukang/publications.html

[4]https://github.com/thunlp/nre

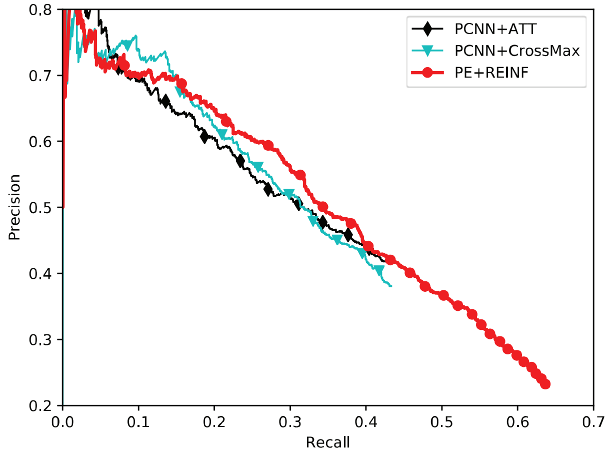

Figure 8: Results of different models on LARGE dataset.

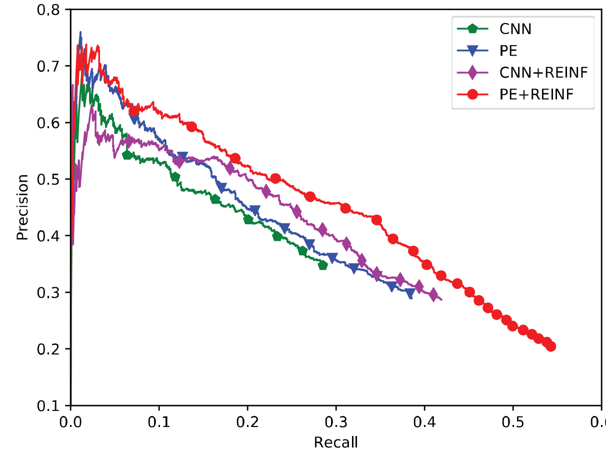

Figure 9: Effects of reinforcement learning in SMALL dataset. By adding the reinforcement learning, we achieve significant improvements in both models.

accuracy. For example, a classifier predicts all sentences to NA relation could achieve about 98% ($\approx 1 - 1732/93574$) accuracy. So we randomly select 100 positive bags which contains 197 sentences.

The baseline extractors are CNN relation extractor and PE relation extractor. These two models are trained under supervised paradigm. That is we label the sentences with the bag relation and treat it as supervised data. Our model, PE+REINF, is PE extractor trained with distant supervised data directly by apply reinforcement learning.

The manually evaluated results are shown in Figure 6. The experiment shows that PE+REINF significantly outperforms the baseline models. Our model gets 19.71% and 13.36% improvement compared with CNN and PE. The improvement from PE to PE+REINF demonstrate that by applying reinforcement learning and using the distant supervision (the bag label) to guide training can lead to better results.

Distant Supervised Relation Extraction

To evaluate our model in large scaled dataset automatically, we apply our model in DSRE task. DSRE task focus on the bag relation extraction and bag relations are directly labeled in distant supervised dataset, which make the automatic evaluation possible.

Baseline methods We compare our model with several baseline methods. Mintz (Mintz et al. 2009) extracted lexical and syntactic features from all sentences. MIML (Surdeanu et al. 2012) is a multi-instance multi-labels model. MultiR (Hoffmann et al. 2011) is a probabilistic, graphic model with multi instance-learning. PCNNs+MIL (Zeng et al. 2015) used PCNN to extract sentence features and applied multi-instance learning. PCNN+CrossMax (Jiang et al. 2016) used the information of all sentences in a Bag by *cross-sentence max-pooling*. APCNNs (Ji et al. 2017) and PCNN+ATT (Lin et al. 2016) both utilized the attention mechanism.

We implement PCNN+CrossMax by ourself by following the settings of (Jiang et al. 2016). We use the results of Mintz, MIML, MultiR, PCNNs+MIL and APCNNs released by (Ji et al. 2017) and the results of PCNN+ATT released by (Lin et al. 2016).

Evaluation Metrics We follow (Riedel, Yao, and McCallum 2010) and evaluate our method with the held-out evaluation. The held-out evaluation compares the predicted relation of the entity pair with the gold relation (which is automatically labeled by Freebase). It's an effective evaluation method for large dataset. We present the results of held-out evaluation with precision/recall curves.

Held-out Evaluation To compare with the baseline methods, we conduct the held-out evaluation on both SMALL and LARGE dataset.

In the SMALL dataset, we compare our method with Mintz, MIML, MultiR, PCNNs+MIL, PCNN+CrossMax and APCNNs. The precision/recall curves of those models are shown in Figure 7. As we can see, our PE+REINF model (the magenta line with circle marker) outperforms all the baseline methods. Though the APCNNs (the brown line with pentagon marker) has higher precision when recall is less than 0.15, our model performs better when the recall is higher. It's also worth mention that the recall of our model reaches about 54%, which is a 46% improvement compared with APCNNs.

In the LARGE dataset, we compare our model (PE+REINF) with PCNN+ATT and PCNN+CrossMax. As the precision/recall curves shown in Figure 8, PE+REINF (the red line with circle marker) outperforms the baseline methods significantly in both precision and recall. The maximum value of recall of our model can reach about 63% while PCNN+ATT (the black line with diamond marker) and PCNN+CrossMax (the blue line with triangle marker) only reach less than 45%. It's about 40% improvement.

We also give out the area under curve (AUC) value of neural network models. In SMALL dataset, the AUC value of PCNs+MIL is 0.180, APCNNs is 0.205, PCNN+CrossMax is 0.169 and our model PE+REINF is **0.252**. In LARGE

Contains(India, Gujarat)	Our Model		DSRE Models
Correction : February 24 , 2007 , Saturday An article in The Arts on Tuesday about the **Indian** film " Parzania , " about a boy who disappeared during violence in **Gujarat** state , where theater owners have refused to show it , referred incorrectly to the reason another film encountered problems being screened in India .	*NA*	*Contains*	*Contains*
Lalji Desai , 38 , left , from Mera , **India** , after pulling to the side of the road during a family vacation -LRB- Mr. Desai is executive director of a nongovernmental organization largely dedicated to helping the Maldharis , a group of seminomadic herdsmen in **Gujarat** state -RRB- : " I am from this same kind of pastoral community , and we 've been working in the remotest area of the Kutch district for a while .	*Contains*		
An article in The Arts on Tuesday about the Indian film " Parzania , " about a boy who disappeared during violence in **Gujarat** state , where theater owners have refused to show it , referred incorrectly to the reason another film encountered problems being screened in **India** .	*NA*		
Mr. Wagoner said G.M. would more than double its production capacity in **India** , to 225,000 vehicles a year , by making more vehicles at its Halol factory in the state of **Gujarat** and building a new plant in Talegaon , Maharashtra , both in western India .	*NA*		
Luce has no patience for the more extreme Hindu nationalism , which seeks to write other identities out of **India** 's history and helps to foment the sort of horrific violence that erupted in **Gujarat** in 2002 , when some 2,000 Muslims were slaughtered while the police looked on -LRB- or collaborated -RRB- .	*NA*		

Figure 10: A real case of extracting the relation of sentences in the bag of *India* and *Gujarat* with our model. We also show the extraction of the bag with our model and DSRE models. As we can see, our model can both extract relation of the single sentence and the bag while DSRE models only extract the relation of the bag. We mark the correct extraction with red color.

dataset, the AUC value of PCNN+CrossMax is 0.260, PCNN+ATT is 0.259 and PE+REINF is **0.336**. Our PE+REINF model achieves the highest AUC value in both datasets.

Effects of Reinforcement Learning To further show the effects of reinforcement learning, we compare the performance of CNN/PE and CNN+REINF/PE+REINF in distant supervised relation extraction task in SMALL dataset.

As Figure 9 shows, the magenta line with diamond marker and the red line with circle marker are the results of CNN+REINF and PE+REINF model respectively, the green line with pentagon marker is the results of CNN extractor, the blue line with triangle is the results of PE. By applying reinforcement learning, CNN+REINF and PE+REINF achieve better results than CNN and PE respectively, which demonstrate the effectiveness of reinforcement learning.

Discussion

In this paper, we propose a novel method with reinforcement learning to learn relation extractor using large scaled distant supervised data.

This task is different from the distant supervised relation extraction (DSRE) task. Our model is trying to extract relations from every single sentence while the DSRE models aim at extracting relation of an entity pair from all sentences that mention these two entities (the bag). Although both our model and DSRE model utilize the distant supervised dataset, we use it in different ways. We use the distant supervision as the guide of reinforcement learning so that we can learn a relation extractor with the large scaled distant supervised dataset and overcome the obstacle of lacking large scaled annotated data. DSRE models are not fit for relation extraction of a single sentence. A real case is shown in Figure 10. As we can see, our extractor extract relations for every single sentence. We can then predict the bag relation based on the relation of sentences in bag. However, the DSRE model only extracts the relation of the bag and regardless of the sentence relation.

Some readers may be confused with our model and PCNN+MIL (Zeng et al. 2015). First of all, PCNN+MIL is a DSRE model which focuses on the relation prediction of the bag, while our model is a relation extraction model which aims to extract the relation of a single sentence. The second difference is we utilize all sentences when training while PCN+MIL only used one sentence. The third difference is the way we predict the relation of the bag. PCNN+MIL directly choose the sentence with the highest probability, however, we select the sentence with the highest probability from none NA sentences.

Conclusion and Future Work

In this paper, we learn the relation extractor with reinforcement learning method on the distant supervised dataset. The bag relation is used as the distant supervision which guides the training of relation extractor. We also apply the relation extractor to help bag relation extraction. Two types of experiments are conducted to evaluate the effectiveness of our model. The experiment results show that our model outperforms comparative baselines significantly.

There are many directions of future work. Most neural models in relation extraction task are based on convolution neural network and utilize position embeddings as the feature. Other neural network structures and features are going to be explored. Another direction is to combine our model with Open Information Extraction (open IE) models. A major problem of open IE models is the extracted relations cannot be linked with KB relations. Our relation extractor can help to make the extracted relation semantically.

Acknowledgements

The research work is supported by the National Science Foundation of China (No. 61533018), the National Basic Research Program of China (No. 2014CB340503) and the National Science Foundation of China (No. 61702512). This work was also supported in part by Beijing Unisound Information Technology Co., Ltd and Huawei Innovation Research Program of Huawei Tech. Ltm.

References

Dai, Z.; Li, L.; and Xu, W. 2016. Cfo: Conditional focused neural question answering with large-scale knowledge bases. In *Proceedings of ACL*, 800–810.

Grishman, R.; Westbrook, D.; and Meyers, A. 2005. Nyus english ace 2005 system description. *Journal on Satisfiability* 51(11):1927 – 1938.

Hendrickx, I.; Kim, S. N.; Kozareva, Z.; Nakov, P.; Ó Séaghdha, D.; Padó, S.; Pennacchiotti, M.; Romano, L.; and Szpakowicz, S. 2009. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the International Workshop on Semantic Evaluations*, 94–99.

Hoffmann, R.; Zhang, C.; Ling, X.; Zettlemoyer, L.; and Weld, D. S. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of ACL*, 541–550.

Ji, G.; Liu, K.; He, S.; and Zhao, J. 2017. Distant Supervision for Relation Extraction with Sentence-Level Attention and Entity Descriptions. In *Proceedings of AAAI*, 3060–3066.

Jiang, X.; Wang, Q.; Li, P.; and Wang, B. 2016. Relation Extraction with Multi-instance Multi-label Convolutional Neural Networks. In *Proceedings of COLING*, 1471–1480.

Kingma, D. P., and Ba, J. L. 2015. Adam: a Method for Stochastic Optimization. In *Proceedings of ICLR*, 1–15.

Li, J.; Monroe, W.; Ritter, A.; Galley, M.; Gao, J.; and Jurafsky, D. 2016. Deep Reinforcement Learning for Dialogue Generation. In *Proceedings of EMNLP*, 1192–1202.

Lin, Y.; Shen, S.; Liu, Z.; Luan, H.; and Sun, M. 2016. Neural Relation Extraction with Selective Attention over Instances. In *Proceedings of ACL*, 2124–2133.

Lin, Y.; Liu, Z.; and Sun, M. 2017. Neural relation extraction with multi-lingual attention. In *Proceedings of ACL*, 34–43.

Luo, B.; Feng, Y.; Wang, Z.; Zhu, Z.; Huang, S.; Yan, R.; and Zhao, D. 2017. Learning with noise: Enhance distantly supervised relation extraction with dynamic transition matrix. In *Proceedings of ACL*, 430–439.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013a. Efficient estimation of word representations in vector space. *CoRR* abs/1301.3781.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; and Dean, J. 2013b. Distributed representations of words and phrases and their compositionality. *NIPS* 3111–3119.

Mikolov, T.; Yih, S. W.-t.; and Zweig, G. 2013. Linguistic regularities in continuous space word representations. In *Proceedings of NAACL*, 296–301.

Mintz, M.; Bills, S.; Snow, R.; and Jurafsky, D. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of ACL*, 1003–1011.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A.; Ostrovski, G.; et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518(7540):529–533.

Narasimhan, K.; Kulkarni, T.; and Barzilay, R. 2015. Language Understanding for Text-based Games Using Deep Reinforcement Learning. In *Proceedings of EMNLP*, 10.

Narasimhan, K.; Yala, A.; and Barzilay, R. 2016. Improving information extraction by acquiring external evidence with reinforcement learning. In *Proceedings of EMNLP*, 2355–2365.

Riedel, S.; Yao, L.; and McCallum, A. 2010. Modeling relations and their mentions without labeled text. In *Proceedings of ECML PKDD*, 148–163.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Den Driessche, G. V.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. 2016. Mastering the game of go with deep neural networks and tree search. *Nature* 529(7587):484–489.

Surdeanu, M.; Tibshirani, J.; Nallapati, R.; and Manning, C. D. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of EMNLP*, 455–465.

Williams, R. J. 1992. Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. *Machine Learning* 8:229–256.

Wu, F., and Weld, D. S. 2010. Open information extraction using wikipedia. In *Proceedings of ACL*, 118–127.

Xu, Y.; Mou, L.; Li, G.; Chen, Y.; Peng, H.; and Jin, Z. 2015. Classifying relations via long short term memory networks along shortest dependency paths. In *Proceedings of EMNLP*, 1785–1794.

Yih, W.; Chang, M.; He, X.; and Gao, J. 2015. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *Proceedings of ACL*, 1321–1331.

Yu, L.; Zhang, W.; Wang, J.; and Yu, Y. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *Proceedings of AAAI*, 2852–2858.

Zeng, D.; Liu, K.; Lai, S.; Zhou, G.; and Zhao, J. 2014. Relation Classification via Convolutional Deep Neural Network. In *Proceedings of COLING*, 2335–2344.

Zeng, D.; Liu, K.; Chen, Y.; and Zhao, J. 2015. Distant Supervision for Relation Extraction via Piecewise Convolutional Neural Networks. In *Proceedings of EMNLP*, 1753–1762.

Zhou, P.; Shi, W.; Tian, J.; Qi, Z.; Li, B.; Hao, H.; and Xu, B. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of ACL*, 207–212.