Improving Relation Extraction with Relation-Based Gated Convolutional Selector

Qian Yi[1,3,*], Guixuan Zhang[1,2], Shuwu Zhang[1,2], and Jie Liu[1,2]

[1] Beijing Engineering Research Center of Digital Content Technology, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2] Advanced Innovation Center for Future Visual Entertainment, Beijing, China
http://www.springer.com/gp/computer-science/lncs
[3] University of Chinese Academy of Sciences, Beijing, China

Abstract. Distant supervision is an effective way to collect large-scale training data for relation extraction. To better solve the wrong labeling problem accompanied by distant supervision, some methods have been proposed to remove noise sentences directly. However, these methods seldom consider the relation label when removing noise sentences, neglecting the fact that a sentence is regarded as noise because the relation it expresses is inconsistent with the relation label. In this paper, we propose a novel method to improve the performance of bag-level relation extractor via removing noise data with a relation-based sentence selector. Specifically, the relation-based gated convolutional unit of the sentence selector can selectively output features related to the given relation, and these features will be used to judge whether a sentence expresses the given relation. The sentence selector is trained with the data automatically labeled by the relation extractor, and the relation extractor improves its performance with the high-quality data selected by the sentence selector. These two modules are trained alternately, and both of them have achieved better performance. Experimental results show that our model significantly improves the performance of the relation extractor and outperforms competitive baseline methods.

1 Introduction

Relation extraction aims to obtain the relationship between two entities from unstructured text. For example, given a sentence *'Donald Trump was born in America.'* and two entities *'Donald Trump'* and *'America'*, relation extraction intends to get the relation *'place of birth'* from them. Earlier works use manually labeled data to train the classifier in a supervised manner and have achieved good performance[16,2,10,17,14]. However, the performances of these models are limited by the scale of the training data, and constructing a large-scale manually labeled dataset is labor consuming. In order to build large-scale dataset automatically, Mintz et al.[9] proposed distant supervision. Distant supervision is based on the idea that if an entity pair (h, t) is contained by a triple (h, t, r) of a given knowledge base, all sentences that contain the entity pair (h, t) will be labeled as the relation r. The h, t, r represent head entity, tail entity and relation, respectively. However, due to the existence of the multi-relational entity pairs, distant supervision suffers from the wrong labeling problem.

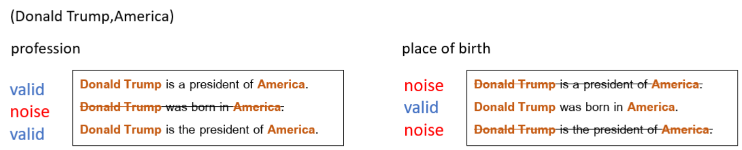


Fig. 1: An example of the noise problem for bag-level relation extraction.

Various methods have been proposed to alleviate this issue. One common way among these studies is to employ Multi-Instance Learning(MIL) schema[13,3], in which sentences containing the same entity pair are divided into the same bag and the classification proceeds on bag-level. Zeng et al.[18] selected the most important sentence to represent the bag and trained the model with these selected sentences. Lin et al.[6] applied attention mechanism to give the important sentences lager weights and combined all sentences to obtain the bag representation. Jiang et al.[5] used cross-sentence max-pooling to find the most prominent features among all sentence representations. Recently, some researchers suggested that it was not enough to attenuate the effects of noise data through 'soft' means like attention mechanism. They tended to remove the noise data directly. Feng et al.[1] and Qin et al.[11] trained a sentence selector to distinguish between noise sentences and valid sentences through reinforcement learning(RL). Qin et al.[12] trained a generative adversarial network(GAN) and used the classifier to remove the noise data.

However, these 'hard' methods neglect the fact that when we consider a sentence as noise, it means that this sentence expresses a relation inconsistent with its label. Just as Figure 1 shows, for the three sentences *'Donald Trump is a presedent of America.'*, *'Donsld Trump was born in America.'*, *'Donsld Trump is the presedent of America.'* and the entity pair *'Donald Trump'* and *'America'*, if the labeling relation is *'place of birth'*, the first and the third sentence are noise sentences. But when the labeling relation is *'profession'*, the second sentence becomes the noise data. Therefore, we think it is crucial to consider the labeling relation when identifying noise data.

In this paper, we propose a novel method to improve the performance of bag-level relation extractor via removing noise data with a relation-based sentence selector. We design a relation-based gated convolutional network for the sentence selector. The relation-based gated convolutional network has two convolutional components. One acts as a feature extractor to extract the semantic features of the sentence. The other is a relation-based gate, which can select the features related to the given relation. Like the previous models, we encounter the problem of lacking training data for the sentence selector. To deal with this problem, we adopt an easy and reasonable method. We treat each sentence as a bag with only one sentence and use the pre-trained bag-level relation classifier to classify it. A sentence will be labeled as a positive sample if the classification result is identical to its label. Otherwise, the sentence is labeled as a negative sample. This labeling method is consistent with the idea that the label of valid data is the same as the relation it conveys. As for bag-level relation extractor, we adopt two widely used architectures: piecewise convolutional neural networks(PCNN) [18] with attention mechanism and the convolutional neural networks(CNN) with attention mechanism. Moreover, because our model is a generic framework, the relation extractor here can be replaced by any other bag-level relation extractor with different structures. Then we train the bag-level relation extractor and the sentence selector alternately so that their performance can be improved jointly.

The main contributions of this paper can be summarized as follows:

• We design a novel relation-based gated convolutional sentence selector to select valid sentences for distantly supervised relation extraction.

• We propose a framework which can train the sentence selector without manually labeled data and jointly improve the performance of the bag-level relation extractor and the sentence selector.

• Experimental results show that our model significantly improves the performance of the relation extractor and outperforms competitive baseline methods.

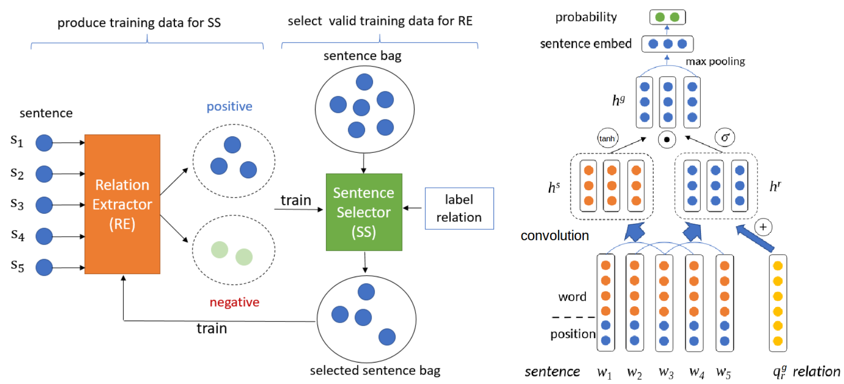
2 Related Work

The purpose of relation extraction is to obtain the relationship between two entities from unstructured text. Traditional methods leveraged syntactic information and adopted kernel-based classifier to build multi-class relation classifier[16,10]. Recently, more attention has been paid to neural networks methods.

In order to extract relation features, previous neural networks models employed various structures to encode the sentence. Zeng et al.[17] adopted CNN to extract the semantic information of the sentence. Xu et al.[15] encoded sentence with Long Short-Term Memory(LSTM) along the shortest dependency path. Zhou et al.[22] combined attention mechanism and LSTM to encode the sentence. Zeng et al.[18] proposed PCNN to extract features from different parts of the sentence separately. Zhang et al.[21] adopted graph convolution over pruned dependency trees to improve the performance of relation extraction. Zhang et al.[19] used attention-based capsule networks to encode the sentence.

In order to solve the problem of lacking for manual annotation data, distant supervision was proposed [9]. To deal with the accompanying wrong labeling problem, researchers have proposed various methods. Zeng et al.[18] adopted the MIL framework. They collected all the sentences containing the same entity-pair as a bag and selected the most important sentence in each bag to train the network. Lin et al.[6] used attention mechanism to give each sentence an importance weight and combined all the sentences to represent the bag. Jiang et al.[5] used cross-sentence max-pooling to extract the features of a sentence bag. Liu et al.[7] softly revised incorrect bag labels with the posterior probability constraint. The above works focused on highlighting the valid sentences of the sentence bag and reduce the effects of noise.

However, some researchers suggested that it was not enough to only weaken the effects of noise data by giving them a small weight, they tended to remove the noise data directly. Feng et al.[1] and Qin et al.[11] trained a sentence classifier to distinguish between noise sentences and other sentences through RL. Qin et al.[12] trained a generative adversarial network and used the classifier to remove the noise data.



(a) Overall framework (b) The structure of the sentence selector

Fig. 2: The architecture of our model.

3 Model

An overview of our framework is shown in Figure 2(a). The model consists of two parts: a relation extractor and a sentence selector. The sentence selector is trained with data automatically labeled by the relation extractor. And the relation-based selector selects data from each sentence bag according to the labeling relation. The relation extractor is then further trained with the selected high-quality sentence bags. These two modules help each other to obtain better training data and finally achieve better performance. In this section, we will first describe these two parts in detail, and then introduce the specific details of training and test.

3.1 Input Layer

Given a sentence s, the input layer transforms the sentence into an embedding matrix, which contains both semantic information and positional information of each word, and feed it to the subsequent networks.

Embedding Word embeddings are low dimensional, continuous and real-valued vectors, which can capture semantic meanings of words. Each word in the vocabulary corresponds to a word embedding vector $v_w \in \mathcal{R}^{d_w}$. In this paper, we use word embeddings pre-trained on the New York Times(NYT) corpus with Skip-Gram[8].

Position Embedding Position embeddings are vectors that embed the relative distances of each token to the two target entities. For example, in the sentence "$SteveJobs$ was the co-founder and CEO of $Apple$ and...", the relative position from token $co-founder$ to entity $SteveJobs$ and $Apple$ is 3 and -4, respectively. Each relative position value corresponds to a position embedding vector $v_p \in \mathcal{R}^{d_p}$.

For each word w, we concatenate its word embedding v_w and two position embeddings(each corresponds to the relative distance from one entity) $v_{p_{en1}}$ and $v_{p_{en2}}$ as its representation $v \in \mathcal{R}^{d_w+d_p*2}$. Then for each sentence with n words $s = \{w_1, ..., w_n\}$, we obtain an embedding matrix $\boldsymbol{S} = \{v_1; ...; v_n\}$ by concatenating all words representations.

3.2 Relation Extractor

Given an entity pair (h, t) and its sentence bag $S_{h,t} = \{s_1, s_2, ...\}$, the relation extractor intends to obtain the relation of the bag. Because our model is a generic framework, the relation extractor module can use any bag-level relation extractor. In this paper, we adopt two widely used models: CNN with attention mechanism and PCNN with attention mechanism.

Sentence Encoder We use CNN and PCNN as sentence encoder to encode the sentence embedding matrix into a representation vector.

Convolution A filter $W_r \in \mathcal{R}^{k_h \times d_v \times m}$ is applied to extract local features of a sentence, where $d_v = d_w + 2 * d_p$ is the dimension of the word vector, m is the width of the filter and k_h is the dimension of the output channel. By sliding W_r along the sentence embedding matrix $\boldsymbol{S}_i$, we can get the k_h-dimensional feature vector:

$$h_i = ReLU([v_{i-(m-1)/2}, ..., v_{i+(m-1)/2}] \otimes W_r + b_r) \tag{1}$$

where $b_r \in \mathcal{R}^{k_h}$ is a bias. Then all the feature vectors are concatenated to form a feature map $H = \{h_1, ..., h_n\}$.

Max-pooling and Piece-wise Max-pooling Max-pooling operation is then applied over the feature map H to get the final sentence representation:

$$q_j = \max_{1 \leqslant i \leqslant n} \{h_{i,j}\} \tag{2}$$

Piece-wise Max-pooling is an extension of Max-pooling:

$$\begin{aligned} q^{(1)}{}_j &= \max_{1 \leqslant i \leqslant i_{en1}} \{h_{i,j}\} \\ q^{(2)}{}_j &= \max_{i_{en1} < i \leqslant i_{en2}} \{h_{i,j}\} \\ q^{(3)}{}_j &= \max_{i_{en2} < i \leqslant n} \{h_{i,j}\} \end{aligned} \tag{3}$$

where the subscript j represents the j-th value of a vector, i_{en1} and i_{en2} are the positions of two entities. Then we concatenate the three pooling vectors to get the final sentence representation:

$$q = \{q^{(1)}, q^{(2)}, q^{(3)}\}. \tag{4}$$

Sentence Selective Attention After obtaining sentence representation, we apply selective attention to compute the attention score α_i for each sentence. Then the bag embedding u is computed as a weighted sum of sentence representations:

$$u = \sum_{i}^{|S_{h,t}|} \alpha_i q_i \tag{5}$$

where the weight α_i indicates the degree of correlation between sentence and the relation, and $|S_{h,t}|$ is the number of sentences in the bag. We assign a query vector $q_r \in \mathcal{R}^{d_q}$ for each relation r, where d_q is the dimension of sentence representation q. The attention score is computed as:

$$\begin{aligned} e_i &= q_r^{\mathrm{T}} W_a q_i \\ \alpha_i &= \frac{\exp(e_i)}{\sum_j^N \exp(e_j)} \end{aligned} \tag{6}$$

where $W_a \in \mathcal{R}^{d_q \times d_q}$ is the weight matrix and N is the number of relations.

Loss Function Finally, we obtain the conditional probability $p(r|S_{h,t}, \theta)$ through feeding the bag representation u to a fully connected layer:

$$\begin{aligned} p(r|S_{h,t}, \theta) &= \frac{\exp(o_r)}{\sum_k \exp(o_k)} \\ o &= W_c u + b_c \end{aligned} \tag{7}$$

where $W_c \in \mathcal{R}^{N \times d_q}$ is a weight matrix and $b_c \in \mathcal{R}^N$ is a bias vector.

Given the collection of sentence bags $\Omega = \{S_{h_1,t_1}, S_{h_2,t_2}, ...\}$ and corresponding labeling relation $\{r_i, r_2, ...\}$, the loss function is defined as follows:

$$J_R = -\frac{1}{|\Omega|} \sum_{i=1}^{|\Omega|} \log p(r_i|S_{h_i,t_i}, \theta) \tag{8}$$

where $|\Omega|$ is the number of bags.

3.3 Sentence Selector

The sentence selector is a binary classifier which can judge whether a sentence expresses the given relation. Its detailed structure is shown in Figure 2(b).

Obtaining Training Data Since there is no training data for the sentence selector, we propose a method to label data automatically. We transform the relation extractor introduced in Section 3.2 into a sentence-level relation extractor by regarding each sentence as a bag with only one sentence. Those sentences whose classification result is consistent with their labeling relation are labeled as positive. Otherwise, they are labeled as negative.

After that, we get a data set $D = \{s, e, y, r\}$, in which s represents the sentence, e is the entity pair, y is the two-category label and r is the labeling relation obtained by distant supervision.

Relation-Based Gated Convolutional Network Given a sentence s_i and its corresponding relation r_i, the input to the relation-based gated convolutional network is the same as the input embedding matrix in Section 3.1. Specifically, each token is embedded into a word embedding and two position embeddings, so that we get the input embedding matrix $\boldsymbol{S}_i = \{v_1; ...; v_n\}$.

Then we feed the input embedding matrix $\boldsymbol{S}_i$ to the relation-based gated convolutional unit. The relation-based gated convolutional unit contains two convolutional components. One is a plain convolution operation mentioned in Section 3.2:

$$h_i^s = tanh([v_{i-(m-1)/2}, ..., v_{i+(m-1)/2}] \otimes W_s + b_s) \tag{9}$$

where $W_s \in \mathcal{R}^{k_h \times d_v \times m}$ is the filter, and $b_s \in \mathcal{R}^{k_h}$ is a bias. The other convolution operation integrates relation information when extracting the local features:

$$h_i^r = \sigma([v_{i-(m-1)/2}, ..., v_{i+(m-1)/2}] \otimes W_g + W_q q_r^g + b_g) \tag{10}$$

where σ is the *sigmoid* function, $W_g \in \mathcal{R}^{k_h \times d_v \times m}$ is the filter and $b_g \in \mathcal{R}^{k_h}$ is a bias, $W_q \in \mathcal{R}^{d_v \times d_q}$ is the weight matrix, q_r^g is the query vector which contains information related to relation r_i. It is worth noting that we initialize the query vector q_r^g with the value of q_r trained in the relation extractor when training the sentence selector. Then we compute an element-wise multiplication between the feature vector h^s and the relation gate vector h^r:

$$h_i^g = h_i^s \odot h_i^r \tag{11}$$

where the symbol $\odot$ represents the element-wise multiplication. Through the element-wise multiplication, the relation gate can selectively output the sentence features related to the given relation. The max pooling procedure is then performed over the feature maps to obtain the sentence embedding:

$$q^g{}_j = \max_{1 \leqslant i \leqslant n} \{h_{i,j}^g\} \tag{12}$$

Loss Function We feed the sentence embedding q_i^g to a fully connected layer to compute the posterior probability $p(y'|s, r, \phi)$:

$$\begin{gathered} p(y'|s, r, \phi) = \frac{\exp(o_{y'}^g)}{\sum_k \exp(o_k^g)} \\ o^g = W_o q^g + b_o \end{gathered} \tag{13}$$

where $W_o \in \mathcal{R}^{2 \times d_q}$ is a weight matrix, $b_o \in \mathcal{R}^2$ is a bias vector and y' is the two-category label.

Given the collection of sentences $\Lambda = \{s_1, s_2, ...\}$, its label relation $\{r_i, r_2, ...\}$ and the corresponding label $\{y_i, y_2, ...\}$, the loss function is defined as follows:

$$J_S = -\frac{1}{|\Lambda|} \sum_{i=1}^{|\Lambda|} \log p(y_i|s_i, r_i, \phi) \tag{14}$$

where $|\Lambda|$ is the number of bags.

Algorithm 1 Overall Training Procedure

Input: Episode number L, collection of sentence bags $B = \{S_1, S_2, ..., S_n\}$, collection of sentences $S = \{s_1, s_2, ..., s_m\}$

Initialize the relation extractor and the sentence selector randomly.

```
for l = 1 to L do
    if Not converge then
        Train the relation extractor with B.
        Label S with the relation extractor to obtain S_label.
        if l = 1 then
            Initalize the relation query matrix Q^g = {q^g_{r_1}; q^g_{r_2}; ...; q^g_{r_N}} with the relation query
            matrix Q = {q_{r_1}; q_{r_2}; ...; q_{r_N}} of the relation extractor.
        end if
        Train the sentence selector with S_label.
        Select the sentences in each sentence bags with the sentence selector to obtained B =
        B_selected.
    else
        End training.
    end if
end for
```

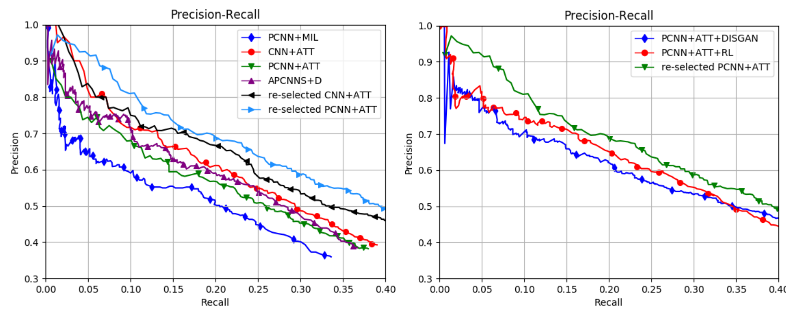
3.4 Training and Test

Because the performance of the relation extractor and the sentence selector influence each other. We train the two modules alternately.

During training, we first adopt Adam algorithm to minimize the loss function Eq. 8 with the original dataset. After using relation extractor to generate training data for the sentence selector, we then optimize the sentence selector by minimizing the Eq. 14 with Adam. Next, we utilize the sentence selector to select the sentences to further train the relation extractor. The relation extractor and the instance selector are trained alternately as described above until convergence. The complete training process is described in Algorithm 1.

During testing, we first select the test data with the sentence selector. Because there is no label for sentence bags during the test procedure, for each sentence bag $S_{h,t}$, we generate one relation-based sentence bag $S^r_{h,t}$ for each relation r by selecting valid sentences from $S_{h,t}$ given relation r. We will obtain N relation-based sentence bags for each sentence bag, where N is the number of relations. Then we feed all N relation-based sentence bags into the relation extractor, and the relation r whose bag obtains the highest score $p(r|S^r_{h,t}, \theta)$ is chosen as the predicted relation for the entity pair (h, t).

What needs to be mentioned is that, in order to increase the recall, we set a threshold $u < 0.5$ when selecting data, and only sentences whose selecting probability is smaller than u will be removed. This operation increases the tolerance for classification errors of sentence selector and enhances the recall of the relation extractor.



(a) Comparison with baseline models (b) Comparison with selector-based models

Fig. 3: Comparison with previous methods.

Table 1: Comparison of P@N between our model and the original bag-level relation extractor

P@N	100	200	500	Mean
CNN+ATT	0.73	0.68	0.56	0.657
re-selected CNN+ATT	**0.79**	**0.76**	**0.60**	**0.717**
PCNN+ATT	0.80	0.71	0.59	0.700
re-selected PCNN+ATT	**0.86**	**0.76**	**0.64**	**0.753**

4 Experiments

4.1 Dataset and Evaluation

In this paper, we evaluate our model on the widely used New York Times(NYT) dataset developed by [13]. This dataset is constructed by aligning Freebase with New York Times(NYT) corpus through distant supervision. There are 522611 sentences in the training set and 172448 sentences in the test set, and these sentences are labeled by 53 candidate relations. Among the 53 relations, there is a label NA, which represents there is no relation between the two entities in a sentence. During training, We randomly extract ten percent of the sentences from the training data as the validation data and the rest as the training data.

We evaluate all methods with the held-out evaluation. The held-out evaluation compares the relational facts extracted from the test set by models with all the facts existing in the test sentences(which is labeled by Freebase through distant supervision). For evaluation, we present precision-recall curves for all models.

4.2 Implementation Detail

In our experiment, our parameter settings are as follows: the dimension of word embedding d_w and position embedding d_p are 50 and 5; the width of the convolution kernel m is 3 and the dimension of the output channel k_h of the convolution filter is 230; the max sentence length is 120; the batch size is fixed to 50 and dropout probability is fixed to 0.5. We adopt Adam to update the parameters, and the learning rate for training relation extractor and sentence selector are set to 0.001 and 0.0005. As for the threshold u for the sentence selector during selecting, we tune it on the validation dataset and pick $u = 0.3$ in the candidate set $\{0.1, 0.2, 0.3, 0.4, 0.5\}$.

4.3 Comparison with Previous Methods

To evaluate the performance of our proposed model, we compare our model with various baseline models. **PCNN+MIL** [18] proposed piecewise CNN to encode the sentence and adopted the MIL framework. **CNN + ATT** and **PCNN+ATT**[6] employed attention mechanism to reduce the influence of noise data and used CNN and PCNN as sentence encoder respectively. **APCNNS+D** [4] used external entity descriptions and attention mechanism to obtain better bag representation. We also compare the performance of our model with other selector-based models to further assess the sentence selector. **PCNN+ATT+DSGAN** [12] trained a generative adversarial network and used the classifier to remove the noise data. **PCNN+ATT+RL** [1] trained a sentence selector through reinforcement learning.

As shown in Figure 3(a), our models, which are denoted as re-selected+PCNN+ATT and re-selected+CNN+ATT, have a significant improvement on all baseline models. For a more detailed comparison, we show the precision@N(P@N) of our models(re-selected+PCNN+ATT and re-selected+CNN+ATT) and the corresponding baseline models(CNN+ATT and PCNN+ATT) in Table 1. The results demonstrate the effectiveness of the sentence selector for distant supervision relation extraction. The re-selected CNN+ATT model and the re-selected PCNN+ATT both achieve higher values for P@100, P@200, P@500 compared to the original baseline models. Moreover, the mean value of re-selected+CNN+ATT is 6% higher than CNN+ATT, and re-selected+PCNN+ATT is 5.3% higher than PCNN+ATT.

As Figure 3(b) shows, when compared to other selector-based models, PCNN+ATT with relation-based gated selector also achieves better performance on both precision and recall. Moreover, compared to the RL and GAN, our model is more stable and easier to converge when training the sentence selector.

4.4 Effect of Relation-Based Convolutional Gate Unit

To further show the effectiveness of the relation-based gated convolutional unit and the necessity of relation information, we compare the performance of three sentence selector architectures: the CNN selector which removes the gate component h_i^r, the plain gated-CNN selector which removes the relation information component $W_q q_r^g$ from Eq. 10 and the relation-based gated-CNN selector.

As Figure 4 shows, the performance of the relation extractors decline drastically when trained with the sentences selected by the plain gated-CNN selector and the CNN selector. The experimental results indicate that the sentence selector is unable to identify the noise sentences with only the semantic features extracted by the CNN or the gated CNN. The good performance of the sentence selector was brought by the relation-based gate which can integrate relation information when extracting the sentence features.

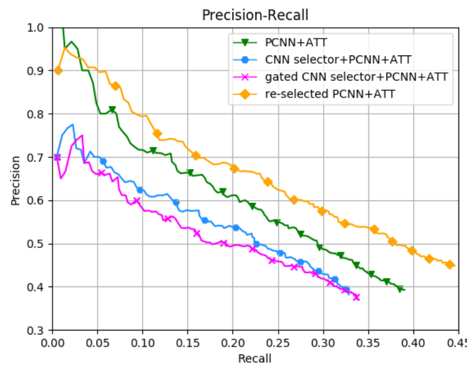


Fig. 4: Comparison between CNN selector, plain gated-CNN selector and relation-based gated-CNN selector.

5 Conclusion

In this paper, we propose a novel method to improve the performance of bag-level relation extractor via removing noise data with a relation-based sentence selector for neural relation extraction under the distant supervision scenario. The whole model contains a relation extractor and a sentence selector composed of a well-designed relation-based gated convolutional network. We train the sentence selector without manually labeled data and employ the selector to select high-quality data for training the relation extractor. We conduct experiments on a widely used dataset. The experimental results confirm the effectiveness of the relation-based gated convolutional unit and our framework significantly improves the performance of the original bag-level relation extractor.

References

1. Feng, J., Huang, M., Zhao, L., Yang, Y., Zhu, X.: Reinforcement learning for relation classification from noisy data. In: Proceedings of AAAI. (2018)
2. GuoDong, Z., Jian, S., Jie, Z., Min, Z.: Exploring various knowledge in relation extraction. In: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics. pp. 427-434(2005)
3. Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., Weld, D.: Knowledge-based weak supervision for information extraction of overlapping relations. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics. pp. 541-550(2011)
4. Ji, G., Liu, K., He, S., Zhao, J.: Distant supervision for relation extraction with sentence-level attention and entity descriptions. In: Proceedings of AAAI. (2017)
5. Jiang, X., Wang, Q., Li, P., Wang, B.: Relation extraction with multi-instance multi-label convolutional neural networks. In: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics. pp. 1471-1480(2016)
6. Lin, Y., Shen, S., Liu, Z., Luan, H., Sun, M.: Neural relation extraction with selective attention over instances. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. pp. 2124-2133(2016)

7. Liu, T., Wang, K., Chang, B., Sui, Z.: A soft-label method for noise-tolerant distantly supervised relation extraction. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pp. 1790-1795(2017)
8. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems. pp. 3111-3119(2013)
9. Mintz, M., Bills, S., Snow, R., Jurafsky, D.: Distant supervision for relation extraction without labeled data. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP. pp. 1003-1011(2009)
10. Mooney, R., Bunescu, R.: Subsequence kernels for relation extraction. In: Advances in neural information processing systems. pp. 171-178(2006)
11. Qin, P., Xu, W., Wang, William Y.: Robust Distant Supervision Relation Extraction via Deep Reinforcement Learning. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. pp. 2137-2147(2018)
12. Qin, P., Xu, W., Wang, William Y.: DSGAN: Generative Adversarial Training for Distant Supervision Relation Extraction. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. pp. 496-505(2018)
13. Riedel, S., Yao, L., McCallum, A.: Modeling relations and their mentions without labeled text. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 148-163(2010)
14. Santos, C., Xiang, B., Zhou, B.: Classifying relations by ranking with convolutional neural networks In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. pp. 626-634(2015)
15. Xu, Y., Mou, L., Li, G., Chen, Y., Peng, H., Jin, Z.: Classifying relations via long short term memory networks along shortest dependency paths. In: Proceedings of the 2015 conference on empirical methods in natural language processing. pp 1785-1794(2015)
16. Zelenko, D., Aone, C., Richardella, A.: Kernel methods for relation extraction. J. Journal of machine learning research. 3, 1083-1106(2003)
17. Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J.: Relation classification via convolutional deep neural network. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics. pp. 2335-2344(2014)
18. Zeng, D., Liu, K., Chen, Y., Zhao, J.: Distant supervision for relation extraction via piecewise convolutional neural networks. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. pp. 1753-1762(2015)
19. Zhang, N., Deng, S., Sun, Z., Chen, X., Zhang, W., Chen, H.: Attention-based capsule networks with dynamic routing for relation extraction. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 985-992(2018)
20. Zhang, Y., Zhong, V., Chen, D., Angeli, G., Manning, C.: Position-aware attention and supervised data improve slot filling. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pp. 35-45(2017)
21. Zhang, Y., Qi, P., Manning, C.: Graph convolution over pruned dependency trees improves relation extraction. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 2205-2215(2018)
22. Zhou, P., Shi, W., Tian, J., Qi, Z., Li, B., Hao, H., Xu, B.: Attention-based bidirectional long short-term memory networks for relation classification. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. pp.207-212(2016)