Distantly Supervised Relation Extraction using Multi-Layer Revision Network and Confidence-based Multi-Instance Learning

Xiangyu Lin[1], **Tianyi Liu**[2], **Weijia Jia**[2,3*] **and Zhiguo Gong**[1*]
[1]SKL-IOTSC and Department of Computer and Information Science, University of Macau[†]
[2]Department of Computer Science and Engineering, Shanghai Jiao Tong University
[3]BNU-UIC Institute of AI and Future Networks, Beijing Normal University (Zhuhai)
yc07403@um.edu.mo, liutianyi@sjtu.edu.cn, jiawj@sjtu.edu.cn, fstzgg@um.edu.mo

Abstract

Distantly supervised relation extraction is widely used in the construction of knowledge bases due to its high efficiency. However, the automatically obtained instances are of low quality with numerous irrelevant words. In addition, the strong assumption of distant supervision leads to the existence of noisy sentences in the sentence bags. In this paper, we propose a novel Multi-Layer Revision Network (MLRN) which alleviates the effects of word-level noise by emphasizing inner-sentence correlations before extracting relevant information within sentences. Then, we devise a balanced and noise-resistant Confidence-based Multi-Instance Learning (CMIL) method to filter out noisy sentences as well as assign proper weights to relevant ones. Extensive experiments on two New York Times (NYT) datasets demonstrate that our approach achieves significant improvements over the baselines.

1 Introduction

Relation Extraction (RE), which aims to classify the relations between a pair of entities in a sentence, is crucial to various applications like question-answering and construction of knowledge bases. However, supervised relation extraction requires large amounts of manually labeled training data, which is hard to obtain. Therefore, Mintz et al. (2009) proposed Distantly Supervised Relation Extraction (DSRE) to automatically generate training data by aligning the knowledge base with text corpus. However, DSRE is based on the strong assumption that for an entity pair participating in a relation in the knowledge base, all sentences mentioning this entity pair in the corpus express the same relation. This brings a large number of noisy sentences into the generated data. The worst case is the noisy bag problem where all the sentences in the bag are mislabeled. On the other hand, low-quality sentences in the corpus contain a large proportion of irrelevant words, meaning even correctly labeled sentences may be filled with inner-sentence noise. To better present the impact of both sentence-level noise and word-level noise (inner-sentence noise), we select a sentence bag from New York Times (NYT) corpus as shown in Figure 1. Among the three sentences, only *S2* expresses the label relation, meaning *S1* and *S3* are both noisy sentences. What's worse, in *S2*, the relation is indicated by a single word *co-founders* and the rest of the words can be regarded as noise.

Entity1: dreamworks Entity2: steven_spielberg
Relation: /business/company/founders

S1	with mr. eastwood as director and **steven_spielberg** as a producer , ... and ferocious backing from paramount and **dreamworks** .
S2	it is hard to say ... the fund-raiser : the **dreamworks** co-founders david geffen , jeffrey katzenberg and **steven_spielberg** .
S3	the outsize robot adventure movie was born ... with **dreamworks** , paramount and another longtime associate , **steven_spielberg** , among others .

Figure 1: An instance from NYT corpus along with its corresponding entity pair and relation type. Relevant words are underlined.

To tackle sentence-level noise, various multi-instance learning (Riedel et al., 2010) methods are proposed to reduce the effects of noisy sentences. Some methods filter out noisy sentences and keep the relevant ones (Zeng et al., 2015; Qin et al., 2018; Feng et al., 2018), but they may filter out relevant sentences as well. Some other methods apply soft labels or weights to limit the impact of noisy sentences (Lin et al., 2016; Liu et al., 2017; Yuan et al., 2019a), yet still at risk of being influenced by sentence-level noise because of the soft

*Corresponding authors
†Xiangyu Lin and Zhiguo Gong are with the State Key Laboratory of Internet of Things for Smart City, Department of Computer and Information Science, University of Macau

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 165–174
November 7–11, 2021.

method. Therefore, a more balanced multi-instance learning strategy should be designed to avoid noisy sentences as well as fully exploit information in relevant ones.

To address word-level noise, a robust encoder is needed for capturing relevant information in a noisy context. Most of the previous work uses encoders based on Recurrent Neural Network (RNN) or Convolutional Neural Network (CNN) (Zeng et al., 2015; Zhou et al., 2016). However, both RNN and CNN models have shortcomings. In RNN encoders, irrelevant words can easily spread noise to its context since they are not isolated from relevant ones. As for CNN encoders, they may lose the salient information because of their pooling layers. To sum up, previous methods are not able to handle noisy words without information loss. Based on the idea that noisy words have weaker correlations with others, we try to fill this gap by modeling the correlations within sentences using attention mechanism.

We propose a novel Multi-Layer Revision Network (MLRN) with Confidence-based Multi-Instance Learning (CMIL) to tackle both word-level and sentence-level noise. In MLRN, we employ the novel revision layers, which alleviate noise by emphasizing inner-sentence correlations, to extract relevant information from the sentence. In each revision layer, we first model the correlations between words using self-attention, then emphasize the correlations by revising the attention weights, and finally apply a Translation Query (TRQ) for information extraction. By stacking multiple revision layers, implicit correlations between words are addressed. To alleviate sentence-level noise and tackle the noisy bag problem, we devise a confidence vector to measure the relevance of sentences to the relation classes and further utilize it to guide sentence filtering and weighting. Our contributions can be summarized as follows:

- To our best knowledge, MLRN is the first model to utilize implicit correlations between words and the first DSRE network based solely on attention mechanism without RNN/CNN encoder layer or extra linguistic information.
- We propose a confidence-based multi-instance learning strategy that is able to (1) conduct sentence filtering independent of DS label to address noisy sentences, and (2) assign proper weights to relevant sentences based on their relevance to the bag prediction.
- Extensive experiments show that our approach achieves significant improvements over the baselines.

2 Related Work

Distant supervision (DS) for relation extraction (Mintz et al., 2009) is proposed for efficient knowledge base construction. However, DS brings about the wrong labeling problem as well. Riedel et al. (2010) proposes multi-instance learning for DSRE to address this issue. Most of the current work uses two types of MIL strategies: to remove noisy sentences or to apply soft weights. Following the at-least-one assumption, Zeng et al. (2015) selects the instance with the highest probability within the bag. Qin et al. (2018) and Feng et al. (2018) employ reinforcement learning for instance selection. For better information utilization, Lin et al. (2016) applies selective attention on the sentence level to dynamically adjust the attention weights. Yuan et al. (2019a) calculates the similarity between instances and the best sentence. DS may create noisy bags where all the sentences are mislabeled. Yuan et al. (2019b) and Ye and Ling (2019) use bag-level attention to address this issue.

CNN-based (Zeng et al., 2015) and RNN-based (Zhou et al., 2016) networks are widely used for capturing information within the sentences. In addition, Xu et al. (2015) and Liu et al. (2018) integrate extra linguistic information into the model to address word-level noise. Since attention mechanism has been proved effective for modeling long-range dependencies in the sequence (Vaswani et al., 2017), attention-based models (Wang et al., 2018; Du et al., 2018; Huang and Du, 2019; Zhang et al., 2020) are also introduced into DSRE.

In our work, we further make use of attention mechanism to emphasize correlations between words and devise a balanced confidence-based strategy to address noisy sentences.

3 Methodology

The overall structure of our model is shown in Figure 2. Our model can be divided into three parts: embedding layer, revision network and multi-instance learning layer. In this section, we introduce them respectively.

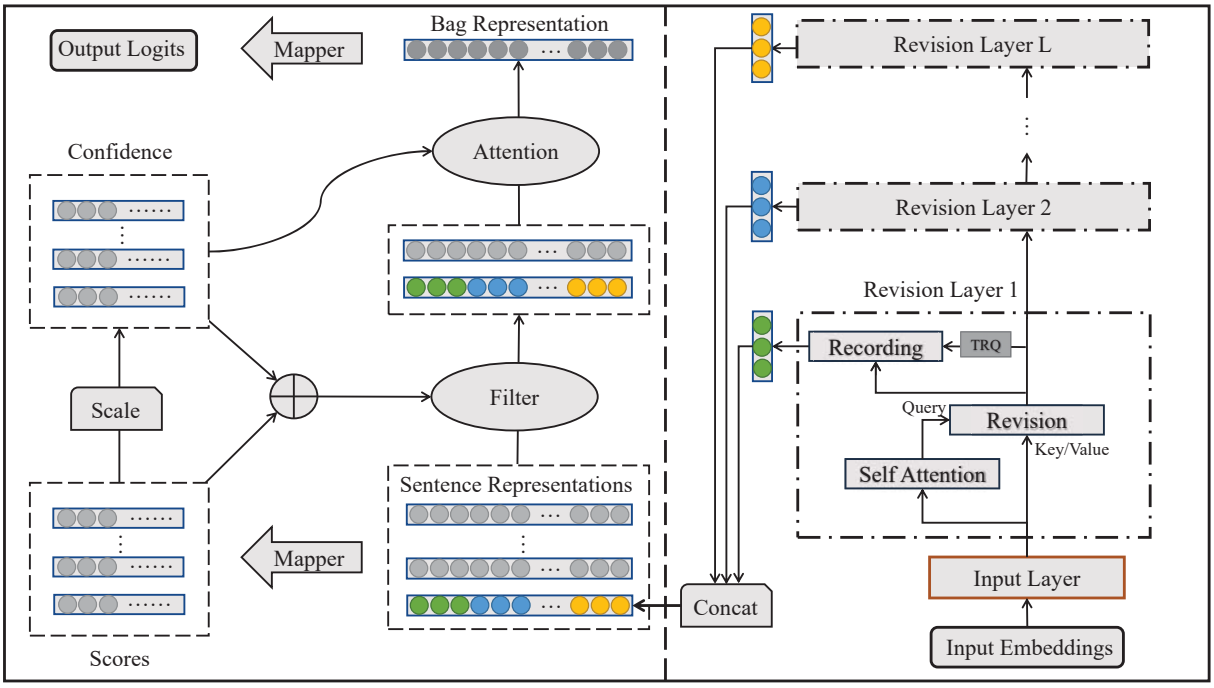

Figure 2: An overview of our MLRN+CMIL model. The revision network used for generating sentence representations is on the right, while confidence-based multi-instance learning is on the left.

3.1 Embedding Layer

Before being fed into the revision network, the input instances are transformed into distributed representations. The representation of each word token consists of two parts: word embedding and position embeddings.

Word Embeddings are distributed representations for word tokens. Formally, we define j_{th} word token in the i_{th} sentence as w_{ij}, which is mapped to a d_w-dimensional word vector $v_{ij} \in R^{d_w}$. The same as previous studies, We adopt Skip-Gram method to obtain the pre-trained word embedding matrix.

Position Embeddings are distributed representations for the relative distances from each word to the two entities, which are represented as low-dimensional vectors $p_{ij}^{e_1}$, $p_{ij}^{e_2} \in R^{d_p}$.

Finally, the input embedding x_{ij} is generated by concatenating word embedding v_{ij}, position embeddings $p_{ij}^{e_1}$ and $p_{ij}^{e_2}$, which is formulated as below:

$$x_{ij} = [v_{ij}; p_{ij}^{e1}; p_{ij}^{e2}] \quad (1)$$

where the dimension of x_{ij} is $d_h = d_w + 2d_p$.

3.2 Revision Network

Formally, the revision network takes a sequence of word representations $X_i = \{x_{i1}, x_{i2}, x_{i3}, ..., x_{il}\}$ with length l as the input, and outputs an d-dimensional representation $U_i^S \in R^d$ for the sentence. The revision layer for word-level noise reduction includes two types of attention sub-layers: self-attention layer and query-attention layer. By applying self-attention on the input, the correlations between each pair of tokens are calculated. In order to emphasize the correlations, the attention weights are revised in a query-attention layer before updating the representations. Afterwards, we apply a Translation Query (TRQ) inspired by TransE (Bordes et al., 2013) to extract relevant information as the record for each layer. Finally, these records are concatenated to form the sentence representation used for multi-instance learning.

The compositions of revision network will be discussed in detail in this section.

3.2.1 Input Layer

The input layer serves as an encoding layer which calculates feature representations from input embeddings. The input is the embeddings of i_{th} instance, denoted as X_i. For convenience, the subscript i is omitted in the equations of this part.

Instead of using CNN or RNN input layers as in most of the previous work, we apply an attention layer to model the long-distance dependencies in the sentence. The attention mechanism used can

be formulated as follows:

$$Att(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (2)$$

where Q is the $query$, K is the key and V is the $value$ as Vaswani et al. (2017). d_k is the dimension of key as well as a scaling factor.

In order to explore various semantic spaces of the sentence, we use Multi-Head Self-Attention (MHSA) in the input layer, which is shown as follows:

$$MHSA(X) = \sigma([H_1; ...; H_h]) \quad (3)$$

$$H_i = Att(XW_i^Q, XW_i^K, XW_i^V) \quad (4)$$

where σ is the sigmoid activation function, $W_i^Q, W_i^K, W_i^V \in R^{hd_k \times d_k}$ are weight matrices for i_{th} head and h is the number of heads. Note that $d_h = hd_k$.

The input layer maps the input embedding onto the feature space, and generates the feature of the instance:

$$S = MHSA(X) \quad (5)$$

The feature S of the instance is then passed to the first revision layer.

3.2.2 Revision Layer

There are totally three steps in revision layer: **Self-Attention**, **Revision** and **Recording**. Formally, the i_{th} input to the k_{th} revision layer is the output from last layer, denoted as S_i^{k-1}. The two entities' representations in the sentence are presented as E_{i1} and E_{i2} respectively. The i_{th} output of k_{th} revision layer is denoted as S_i^k and the record of the k_{th} layer as U^k. For convenience, the superscript k and subscript i are omitted in the following equations except Eq. 10.

Self-Attention layer first calculates the inner-sentence correlations from the input S using self-attention, which is shown as follows:

$$Q^R = Att(S) \quad (6)$$

where we omit repetition of S as we have identical $query$, key and $value$. Note that different from the input layer, the self-attention layer does not introduce extra weight matrices so that the operation is on the same feature space.

Viewed in a word-level perspective, self-attention layer updates the representation of a token as the weighted sum of the representations of all tokens in the sentence, where those similar to the inspected token have larger weights. In the feature space, closely correlated words tend to have similar representations. Since noisy words have weaker correlations with others, their attention weights assigned by other tokens are smaller. Therefore, noisy words are marginalized throughout the process, making them unlikely to spread noise to the rest of the sentence. However, since each token always has the highest similarity with itself, the weights assigned to other relevant words are relatively small, which limits the modeling of inner-sentence correlations. In other words, we need to assign larger weights to relevant words for stronger inner-sentence correlations.

To address this issue, **Revision** process is conducted using a query-attention layer, which takes Q^R, the output from the self-attention layer, as the $query$ and the layer input S as the key and $value$. The calculation is formulated as follows:

$$O = Att(Q^R, S) \quad (7)$$

where we also omit repetition of S since it serves as both key and $value$. We use the output from the self-attention layer as the query because it has already partially modeled the inner-sentence correlations, meaning that in the feature space, relevant words become closer to each other. Therefore, the revision query-attention layer assigns larger weights to relevant words to emphasize the inner-sentence correlations. At the same time, noisy words are further marginalized in this process.

However, not all the words in the sentence are relevant to the relation, as shown in Figure 1. Hence we carry out **Recording** process to extract relevant information from the sentence. In order to represent the relation feature, we employ the TRQ inspired by TransE (Bordes et al., 2013) which uses the difference of two entities' representations as the relation feature. Similar method has been proved effective in Liu et al. (2020). Here, we use multi-head attention to explore multiple semantic sub-spaces, the process is formulated as follows:

$$U = \sigma([H_1; ...; H_h]) \quad (8)$$

$$H_i = Att((E_1 - E_2)W_i^Q, OW_i^K, OW_i^V) \quad (9)$$

where σ is the activation function, E_1 and E_2 are representations of entity pair, W_i^Q, W_i^K, W_i^V are weight matrices for i_{th} head and h is the number of heads. The translation query $E_1 - E_2 \in R^{d_h}$ and the updated representations $O \in R^{l \times d_h}$ are

first mapped onto the same vector space, then the record $U \in R^{d_h}$ is calculated as the weighted sum of the tokens' representations according to their relevance to the relation.

However, implicit correlations may not be considered within a single revision layer. As a simple example, given the sentence *"[Joe] is the father of John, father of [Amy]"*, the model calculates the correlation between *Joe* and *John* as well as the correlation between *John* and *Amy*, but may be unable to observe the correlation between *Joe* and *Amy* in the first layer because they are not directly related. Therefore, we stack multiple revision layers to capture more implicit correlations between the words.

At the end of the revision network, all extracted records are concatenated together to form the final representation for the instance, which is formulated as below:

$$U^S = [U^1; U^2; ...; U^L] \tag{10}$$

where L is the number of revision layers and $U^S \in R^{Ld_h}$ is the final sentence representation. For convenience, in the following sections, we use $d = Ld_h$ to represent the dimension of sentence representations.

3.3 Confidence-based Multi-Instance Learning

After obtaining the representation for each sentence in the bag, we generate the bag representation using the CMIL strategy. First, we filter out noisy sentences according to the prediction of the bag. Afterwards, we emphasize the sentences with higher relevance according to the confidence vector. Formally, the input is a bag of sentence representations: $B = \{U_1^S, U_2^S, ..., U_N^S\}$ where N is the number of sentences in the bag. The output of multi-instance learning layer is the bag representation U^B.

Firstly, as shown in Figure 2, the score of i_{th} sentence in the bag is calculated as follows:

$$F_i = W_r U_i^S + b_r \tag{11}$$

where $W_r \in R^{d \times c}$ and $b_r \in R^c$ are weight matrices and c is the number of relation classes. The confidence vector $C_i \in R^c$ measuring the relevance to each of the relation classes is calculated from the scores as follows:

$$C_{ij} = W_j^c F_{ij} \tag{12}$$

where $W^c \in R^c$ is a weight matrix that represents the reliability of DS labels. In other words, it reflects the model's confidence towards the DS labels. Reliable DS labels have higher possibility to have true positive sentences, therefore, the model becomes more confident towards these labels so they have higher weights in W^c. Afterwards, we obtain the adjusted score, which is the sum of original score and confidence vector, to generate the bag prediction and select relevant instances into the new bag as follows:

$$j^* = argmax_j(C_{ij} + F_{ij}) \quad 1 \le i \le N \tag{13}$$

$$Bag = \{U_i^S | \, argmax_j(C_{ij} + F_{ij}) = j^*\} \tag{14}$$

where $C_{ij} + F_{ij}$ is the adjusted score which applies different thresholds on different relation classes. As shown above, the filtering process is guided by j^*, which is the prediction made by the model. Therefore, in our model, instance selection is guided by the true relation class expressed in the bag instead of being misled by the DS label as in most of the previous methods. In this way, our model is able to alleviate the noisy bag problem.

Finally, in order to obtain the bag representation, we apply weighted sum on the instances in the new bag according to their confidence values:

$$U^B = \sum_i \alpha_i U_i^S \tag{15}$$

$$\alpha_i = softmax(C_{ij^*}) \tag{16}$$

where U_i^S is the representation of i_{th} instance in the new bag and U^B is the final bag representation.

3.4 Optimization

Our goal is to maximize the conditional probability for the target relation given the bag of sentences. The probability $p(y|U^B, \theta)$ is calculated from the bag representation as below:

$$p(y|U^B, \theta) = softmax(W_r U^B + b_r) \tag{17}$$

where W_r and b_r are the same weight matrices as Eq. 11. Then we employ a negative log-likelihood loss function with L_2 regularization to train the model:

$$J(\theta) = -\frac{1}{c} \sum_{k=1}^{c} y_k log(p_k) + \beta ||\theta||^2 \tag{18}$$

where β is a hyper-parameter to restrict the L_2 term. In our work, we use Adam (Kingma and Ba, 2014) to optimize our model.

4 Experiments

In this section, we first introduce the datasets and evaluation metrics used in the experiments. Then, we provide our experimental settings. Afterwards, we compare our model with baselines using the evaluation metrics. Finally, we discuss the effects of the revision layer and the CMIL strategy.

4.1 Datasets and Evaluation Metrics

In order to evaluate the performance of our model, we conduct experiments on widely used NYT-10 dataset (Riedel et al., 2010) and complex NYT-18 dataset (Zhang et al., 2020). NYT-10 is a standard dataset constructed by aligning relation facts in Freebase (Bollacker et al., 2008) with the New York Times corpus, where sentences from 2005 to 2006 are used for training and sentences from 2007 are used as the test set. NYT-18 is a larger dataset containing NYT documents from 2008 to 2017. Both datasets are labeled with Freebase and Stanford Named Entity Recognizer (Finkel et al., 2005). All the sentences are divided into five parts with the same relation distribution for five-fold cross-validation. The details of the datasets are shown in Table 1.

Datasets	Rel.	Training (k)		Testing (k)	
		Ent.	Sen.	Ent.	Sen.
NYT-10	53	281	523	97	172
NYT-18	503	1234	2446	394	611

Table 1: The details of datasets. **Rel.**, **Ent.** and **Sen.** indicate numbers of relations, entity pairs and sentences respectively.

Following previous work (Mintz et al., 2009), we evaluate our model in the held-out evaluation, in which the relations extracted are automatically compared with those in Freebase. PR curves, area under curve (AUC) and Precision at top 100 predictions (P@100) are adopted as the evaluation metrics in our experiments. We employ three test settings which are **One**, **Two** and **All**.

- **One:** For each entity pair, we randomly select one instance to express the relation.
- **Two:** For each entity pair, we randomly select two instances to express the relation.
- **All:** In testing process, all instances mentioning the entity pair are selected.

Parameter	Value
Batch size b	50
Word embedding size d_w	50
Position embedding size d_p	5
Sentence length l	70
Hidden size d_h	60
Number of attention heads h	2
Number of revision layers L	6
Sentence representation size d	360
Learning rate lr	0.0001
Dropout probability pr	0.5
$L2$ penalty β	1e-04

Table 2: Parameter settings.

4.2 Experimental Settings

In the experiments, the word embeddings are pre-trained using word2vec (Mikolov et al., 2013). In Table 2, we list our parameters for the best model. We use two attention heads because we have a pair of mentioned entities in each sentence. The number of revision layers depends on the number of entity pairs and sentences in the dataset, and the difficulty of understanding them. In CMIL process, if all sentences in the bag are filtered, the model will assign average weights to them.

4.3 Evaluation on NYT-10

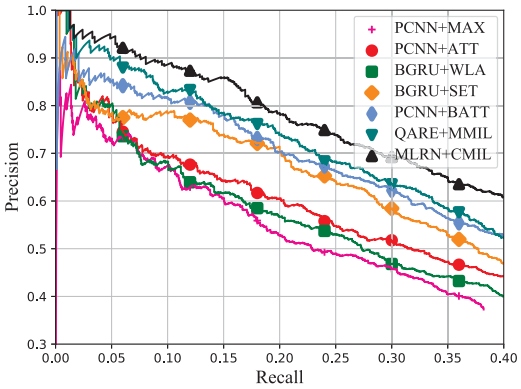

Figure 3: Precision-recall curves of models on NYT-10.

To evaluate our approach, we select the following methods as our baselines:

PCNN+MAX (Zeng et al., 2015) proposes a piecewise CNN model which selects the instance with the largest logit value.

PCNN+ATT (Lin et al., 2016) integrates PCNN

with selective attention mechanism.

BGRU+WLA (Zhou et al., 2016) uses BGRU with word-level attention.

PCNN+RL(Feng et al., 2018) presents a reinforcement learning method for instance selection.

BGRU+SET (Liu et al., 2018) devises a BGRU-based approach to reduce inner-sentence noise.

PCNN+BATT (Ye and Ling, 2019) employs both sentence-level and bag-level attention to emphasize correctly labeled sentences and bags.

QARE+MMIL (Zhang et al., 2020) presents a QA-based relation extractor with transfer learning.

Methods	P@100			
	One	Two	All	mean
PCNN+MAX	73.3	70.3	72.3	72.0
PCNN+ATT	78.0	75.0	82.0	78.3
BGRU+WLA	72.0	70.0	74.0	72.0
PCNN+RL	75.0	79.0	80.0	78.0
BGRU+SET	83.0	85.0	87.0	85.0
QARE+MMIL	87.0	88.0	91.0	88.7
PCNN+BATT	86.8	91.2	91.8	89.9
MLRN+CMIL	**97.0**	**98.0**	**95.0**	**96.7**
Ablations				
PCNN+CMIL	86.0	93.0	92.0	90.3
OneLayer	93.0	93.0	85.0	90.3
SelfAtt+CMIL	88.0	93.0	87.0	89.3
MLRN+MAX	91.0	90.0	90.0	90.3
MLRN+ATT	94.0	97.0	87.0	92.7
MLRN+NIID	95.0	97.0	86.0	92.7

Table 3: P@100 values of the models on NYT-10. **Bold** numbers indicate the best results among all methods. Underlined numbers indicate the best results for CNN/RNN-based models.

Methods	AUC	
	NYT-10	NYT-18
PCNN+MAX	0.216	0.492
PCNN+ATT	0.258	0.511
BGRU+WLA	0.344	0.596
BGRU+SET	0.392	0.290
PCNN+BATT	0.423	0.617
QARE+MMIL	0.428	0.645
MLRN+CMIL	**0.498**	**0.690**

Table 4: AUC of the models on both datasets. **Bold** numbers indicates the best results among all methods.

As shown in Figure 3, MLRN+CMIL outperforms all the baselines significantly without any additional information(e.g. entity types in BGRU+SET). The P@100 values and AUC are shown in Table 3 and Table 4 respectively. Our model improves the AUC to 0.498, which outperforms the best baseline by 16.4%. Moreover, our model achieves the highest P@100 values in all three settings. The results demonstrate that our model can effectively alleviate the influence of both word-level and sentence-level noise.

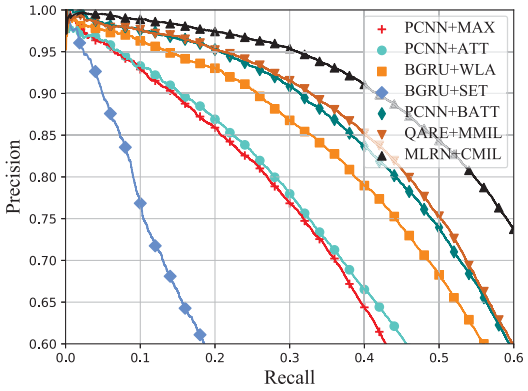

Figure 4: Precision-recall curves of models on NYT-18.

4.4 Evaluation on NYT-18

As presented in Table 4 and Fig 4, our model also significantly outperforms all the baselines on complex NYT-18 dataset. BGRU+SET fails in NYT-18 because the complex instances are difficult to be parsed precisely using the conventional parser. The results prove the robustness of MLRN+CMIL in handling complex instances with inner-sentence noise.

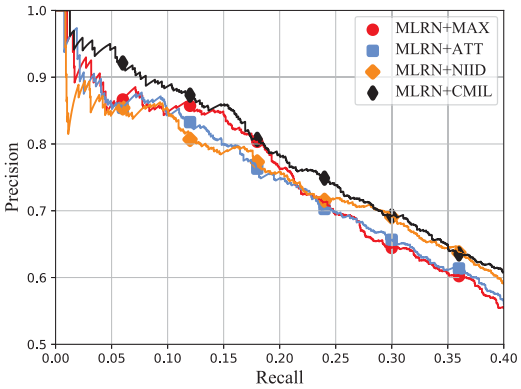

Figure 5: Precision-recall curves of MLRN with various MIL strategies.

Entity1:david_ben-gurion Entity2:israel Relation: /people/person/nationality

Sentences		Relevance	SelfAtt +CMIL	MLRN +ATT	MLRN +CMIL
S1	he said **israel** 's first leader , **david_ben-gurion** , …	Relevant	0.177	0.001	0.352
S2	mr. kollek , a former aide to **david_ben-gurion** , **israel** 's founding prime minister , …	Relevant	0.823	0.999	0.645
S3	mr. bar-zohar ,a noted biographer of **david_ben-gurion** , first wrote … published in **israel** …	Irrelevant	Filtered	0.0	Filtered
S4	mr. feldman … meet quietly with israeli leaders , particularly **david_ben-gurion** …whether **israel** was building …	Relevant (implicit)	Filtered	0.0	0.003

Figure 6: A bag of instances from test data. The numbers indicate the weights assigned in MIL process.

4.5 Ablation Study

In order to evaluate the effects of our approach, we devise the following variations for comparison:

- **PCNN+CMIL**: PCNN network with CMIL.
- **SelfAtt+CMIL**: Removing revision query-attention layer (revision mechanism disabled).
- **OneLayer**: Using only one revision layer.
- **MLRN+MAX**: MLRN which selects the instance with the largest logit value.
- **MLRN+ATT**: MLRN with selective attention.
- **MLRN+NIID**: MLRN with NIID relevance embedding (Yuan et al., 2019a).

As shown in Table 3, MLRN models achieve significant improvements over PCNN models when using the same multi-instance learning strategies (MAX,ATT and CMIL). The complete model outperforms SelfAtt+CMIL significantly, showing that the revision mechanism is crucial for modeling inner-sentence correlations. OneLayer suffers from a dramatic drop in performance because of its incapability in modeling implicit correlations between the words. These results demonstrate that by strengthening correlations in revision process and modeling implicit correlations with multiple revision layers, MLRN becomes more robust and effective in DSRE.

Without bag-level operations in PCNN+BATT, PCNN+CMIL still achieves the highest P@100 mean value among all the PCNN-based models, showing that CMIL can effectively alleviate sentence-level noise and utilize information in the sentence bag. We also test multiple multi-instance learning strategies on MLRN model (MAX, ATT and NIID), and the results in Table 3 and Figure 5 show that CMIL outperforms all of them.

5 Case Study

In Figure 6, we select a bag of instances from test set and present their assigned weights in different models . Among the four sentences, *S1*, *S2* and *S4* are all relevant to the relation. But in *S4*, the relation is indicated in an implicit way by the word "israeli". *S3* is an irrelevant sentence that does not mention the nationality of the entity *david_ben-gurion*.

As the example shows, all the three methods are able to correctly handle the irrelevant sentence *S3*. Although SelfAtt+CMIL works fine when *S1* uses the phrase "israel 's first leader", it wrongly filters out *S4* when encountered with the phrase "israeli leaders". It is because that SelfAtt+CMIL is unable to detect the relevance between "israel" and "israeli" in *S4*. The selective attention method is designed to exploit relevant sentences, but in an attempt to avoid sentence-level noise, it may also down-weight the relevant sentences it has less confidence in, such as *S1* and *S4*. Our complete model (MLRN+CMIL) successfully detects the relevance between "israel" and "israeli", therefore regards $S4$ as a relevant sentence. Moreover, MLRN+CMIL assigns more balanced weights to relevant sentences comparing with other methods.

This example verifies MLRN's ability to capture implicit correlations between sentences. It also proves that CMIL not only alleviates sentence-level noise, but also makes further progress in information utilization.

6 Conclusion and Future Work

In this paper, we propose a novel MLRN+CMIL model for distantly supervised relation extraction. The MLRN structure is able to alleviate noise by modeling inner-sentence correlations and extract relevant information. The CMIL strategy is a balanced and robust way to avoid noisy sentences as well as assign proper weights to relevant ones. The experimental results show that our approach achieves significant improvements over the baselines and is effective in handling both word-level and sentence-level noise.

In the future, we will try to extend our confidence-based method to bag-level, and experiment with the novel revision network on other tasks to further prove its effectiveness.

Acknowledgement

This work was supported by National Key D&R Program of China (2019YFB1600704), The Science and Technology Development Fund, Macau SAR (0068/2020/AGJ, 0045/2019/A1, 0007/2018/A1, SKL-IOTSC-2021-2023), GSTIC (201907010013, EF005/FST-GZG/2019/GSTIC), University of Macau (MYRG2018-00129-FST) and GDST (2020B1212030003).

References

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795.

Jinhua Du, Jingguang Han, Andy Way, and Dadong Wan. 2018. Multi-level structured self-attentions for distantly supervised relation extraction. *arXiv preprint arXiv:1809.00699*.

Jun Feng, Minlie Huang, Li Zhao, Yang Yang, and Xiaoyan Zhu. 2018. Reinforcement learning for relation classification from noisy data. *arXiv preprint arXiv:1808.08013*.

Jenny Rose Finkel, Trond Grenager, and Christopher D Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 363–370.

Yuyun Huang and Jinhua Du. 2019. Self-attention enhanced cnns and collaborative curriculum learning for distantly supervised relation extraction. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 389–398.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv: Learning*.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2124–2133.

Tianyi Liu, Xiangyu Lin, Weijia Jia, Mingliang Zhou, and Wei Zhao. 2020. Regularized attentive capsule network for overlapped relation extraction. *arXiv preprint arXiv:2012.10187*.

Tianyi Liu, Xinsong Zhang, Wanhao Zhou, and Weijia Jia. 2018. Neural relation extraction via inner-sentence noise reduction and transfer learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2195–2204.

Tianyu Liu, Kexiang Wang, Baobao Chang, and Zhifang Sui. 2017. A soft-label method for noise-tolerant distantly supervised relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1790–1795.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. 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*, pages 1003–1011.

Pengda Qin, Weiran Xu, and William Yang Wang. 2018. Robust distant supervision relation extraction via deep reinforcement learning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2137–2147.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Guanying Wang, Wen Zhang, Ruoxu Wang, Yalin Zhou, Xi Chen, Wei Zhang, Hai Zhu, and Huajun Chen. 2018. Label-free distant supervision for relation extraction via knowledge graph embedding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2246–2255.

Yan Xu, Lili Mou, Ge Li, Yunchuan Chen, Hao Peng, and Zhi Jin. 2015. 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*, pages 1785–1794.

Zhi-Xiu Ye and Zhen-Hua Ling. 2019. Distant supervision relation extraction with intra-bag and inter-bag attentions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2810–2819.

Changsen Yuan, Heyan Huang, Chong Feng, Xiao Liu, and Xiaochi Wei. 2019a. Distant supervision for relation extraction with linear attenuation simulation and non-iid relevance embedding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7418–7425.

Yujin Yuan, Liyuan Liu, Siliang Tang, Zhongfei Zhang, Yueting Zhuang, Shiliang Pu, Fei Wu, and Xiang Ren. 2019b. Cross-relation cross-bag attention for distantly-supervised relation extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 419–426.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1753–1762.

Xinsong Zhang, Tianyi Liu, Pengshuai Li, Weijia Jia, and Hai Zhao. 2020. Robust neural relation extraction via multi-granularity noises reduction. *IEEE Transactions on Knowledge and Data Engineering*.

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212.