

Research Article

RTJTN: Relational Triplet Joint Tagging Network for Joint Entity and Relation Extraction

Zhenyu Yang,^{1,2,3} Lei Wang^{ID, 1,2,3} Bo Ma^{ID, 1,2,3} Yating Yang,^{1,2,3}
Rui Dong,^{1,2,3} and Zhen Wang^{1,2,3}

¹The Xinjiang Technical Institute of Physical and Chemistry, Chinese Academy of Sciences, Urumqi 830011, China

²University of Chinese Academy of Sciences, Beijing 100049, China

³Xinjiang Laboratory of Minority Speech and Language Information Processing, Urumqi 830011, China

Correspondence should be addressed to Lei Wang; wanglei@ms.xjb.ac.cn and Bo Ma; mabo@ms.xjb.ac.cn

Received 8 September 2021; Accepted 29 September 2021; Published 16 October 2021

Academic Editor: Wei Xiang

Copyright © 2021 Zhenyu Yang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Extracting entities and relations from unstructured sentences is one of the most concerned tasks in the field of natural language processing. However, most existing works process entity and relation information in a certain order and suffer from the error iteration. In this paper, we introduce a relational triplet joint tagging network (RTJTN), which is divided into joint entities and relations tagging layer and relational triplet judgment layer. In the joint tagging layer, instead of extracting entity and relation separately, we propose a tagging method that allows the model to simultaneously extract entities and relations in unstructured sentences to prevent the error iteration; and, in order to solve the relation overlapping problem, we propose a relational triplet judgment network to judge the correct triples among the group of triples with the same relation in a sentence. In the experiment, we evaluate our network on the English public dataset NYT and the Chinese public datasets DuIE 2.0 and CMED. The F1 score of our model is improved by 1.1, 6.0, and 5.1 compared to the best baseline model on NYT, DuIE 2.0, and CMED datasets, respectively. In-depth analysis of the model's performance on overlapping problems and sentence complexity problems shows that our model has different gains in all cases.

1. Introduction

Joint extraction of entity and relation is an indispensable work for processing unstructured text information and constructing knowledge graphs, which aims to extract all relational triplets in the text. The form of relational triplets is (*subject, relation, object*), for example (Washington, Capital of, America).

Early work used the pipeline extraction method [1, 2] to extract entities and relations separately. But the pipeline method ignores the connection between entity extraction and relation extraction. Therefore, a lot of work in recent years has focused on the joint extraction of entities and relations, like methods based on artificially constructed features [3–5] and neural network methods [6–8]. With the development of deep learning and the application of pre-trained models, entity and relation extraction has reached a

new level of performance [9, 10]. But joint extraction makes the task complicated and many new problems appear, like EntityPairOverlap (EPO), SingleEntityOverlap (SEO), and RelationOverlap (RO) (see Figure 1).

In order to solve the entities and relations overlapping problems, many researchers have proposed solutions. Therefore, many excellent models have emerged to solve the overlapping problems. For example, Zeng et al. [11] proposed a joint entities and relations extraction model based on the seq2seq method; Nayak and Ng [12] applied the encoder-decoder to the entities and relations joint extraction framework; and Wei et al. [8] proposed that relation is a learnable formula from subject to object. These models have excellent results in extraction of entities and relations, but they have a common shortcoming: artificially decomposing extraction of entities and relations into multiple subtasks. They still decompose the extraction of entities and relations

		Text	Triplets
Normal		[The United States] President [Joe Biden] arrives in [The United Kingdom] to participate in the G7 summit in [Cornwall]	(The United States, president, Joe Biden) (The United Kingdom, contains, Cornwall)
SEO		[The United States] President [Joe Biden] was born in [Pennsylvania]	(The United States, president, Joe Biden) (Joe Biden, born_in, Pennsylvania)
EPO		[Washington D.C.] is the political center of [the United States]	(The United States, contains, Washington D.C.) (The United States, capital, Washington D.C.)
RO		[The United States] President [Joe Biden] and [Russian] President [Putin] meet in Geneva	(The United States, president, Joe Biden) (Russian, president, Putin)

FIGURE 1: Examples of the normal, EntityPairOverlap (EPO), SingleEntiyOverlap (SEO), and RelationOverlap (RO). The overlapping entities and relations are masked in red.

into several subtasks in the end-to-end model. The problem with this approach is that the entity extraction task and the relation extraction task are only exchanging a small amount of information through artificial design and the entity extraction task’s error will directly affect the relation extraction.

In this paper, we present a new entities and relations tagging method that can tag all entities and relations information in one label. Our tagging method can turn the extraction of entities and relations model into a simple sequence tagging model and all entities and relations information can be output at one time. Given a sentence, we tag each word with a new label that is divided into 5 regions: the head of the subject (S-H), the tail of the subject (S-T), the head of the object (O-H), the tail of the object (O-T), and the nonentity word region (O). For each region in the label, we are inspired by the binary tagging model to add predefined relations information to each region outside the nonentity word region and use binary 0,1 to represent the discrimination result. This tagging method is a good solution to EPO and SEO problems.

In addition to EPO and SEO problems, our paper formally discusses the RO problem (see Figure 1) for the first time. In the RO problem, multiple relational triplets are sharing the same relation. However, multiple triplets of the same relation have similar contexts, which results in the entity embeddings in the triplets becoming also very similar. Therefore, in the RO problem, models are difficult to identify the subject and object in the same triplet. The RO problem has been ignored by previous work. Most works only use heuristic methods to combine the closest subject and object [11, 13]. For solving RO problem, we propose a relational triplet judgment network to distinguish the subject and object of the same relation.

In summary, this work’s main contributions are as follows:

- (1) We present a joint tagging method for entities and relations in sentence. This method not only enables the complete integration of entity information and relational information but also addresses EPO and SPO problems.
- (2) We evaluate the model using the relation overlapping problem, which is where multiple triplets in

a sentence share the same relation for the first time; and we present a relational triplet judgment network for the RO problem. Our model has achieved good results on the dataset for this problem.

- (3) In the experiment, we evaluate our model on the English dataset NYT [14] and the Chinese datasets DuIE 2.0 and CMED. Our model outperforms the previous models on the three datasets.

2. Related Work

The extraction of relational triplets has always been an important task in the field of natural language processing. It can not only extract knowledge from the unstructured text but also provide prior knowledge for many important tasks of artificial intelligence, like Question Answering System, Knowledge Graph, and Machine Translation.

In early work, the problem of relational triplet extraction is mainly solved by pipeline method, which first identifies all entities in the text and then identifies all relations between any two entities. Nadeau and Sekine [1] and Zelenko et al. [2] used a pipeline to identify entities and relations in the text for the first time; but pipeline method ignores the connection between the two tasks and does not play a role in mutual optimization. To solve this problem, many papers have proposed joint extraction models of entities and relations. Ren et al. [5], Li and Ji [4], and Miwa and Sasaki [15] proposed joint extraction models of entity and relations based on artificially constructed features; but artificial feature construction is difficult and model performance is unstable. Zheng et al. [16] used the LSTM network to realize the joint extraction of entities and relations and used the heuristic principle of proximity to combine the subject and the object to solve RelationOverlap (RO) problem. Zeng et al. [11] proposed that the extraction model of entities and relations needs to solve the overlapping problems, Entity-PairOverlap (EPO) and SingleEntiyOverlap (SEO), at the first time.

In order to solve the entities and relations overlapping problems, many papers have proposed solutions. Fu and Ng [17] added the graph convolutional network to the entities and relations joint extraction model. Nayak and Ng [12] added the idea of encoder-decoder to the model of joint

extraction of entities and relations. Bekoulis et al. [18] transformed the task of extracting entities and relations into a multihead selection task. Yu et al. [19] added a span-based tagging strategy and layered decoding strategy to the joint extraction task. Liu et al. [20] extracted entities based on conditional random fields and judged relation based on supervised multihead self-attention. Wei et al. [8] proposed that relations are constructed as a function from subject to object. Although these methods use end-to-end method to integrate entity extraction and relation extraction into a whole, the entity and relation are divided into several subtasks within the model. Our joint entities and relations tagging method can completely merge entity extraction and relation extraction into one task.

3. Relational Triplet Joint Tagging Network

In this section, we will introduce our relational triplet joint tagging network. First, we will elaborate on joint entities and relations tagging method, which can convert the joint extraction of entities and relations into a sequence tagging problem and output result at one time. Our tagging method can solve the EPO and SEO problems very well. Then we will introduce the relational triplet judgment network for the RO problem, which can integrate sentence information, entity semantic information, and location information to judge whether the triples in the joint tag are correct.

3.1. Joint Entities and Relations Tagging Layer. In order to eliminate the error between entity extraction and relation extraction, we design a joint tagging method of entities and relations to enable the model to output all information at one time instead of processing it separately. We treat the task of entities and relations joint extraction as a sequence tagging task. For a sentence, each word corresponds to a label with entities information and relations information (see Figure 2). For entities information, we divide the label of each word into five regions: the head of the subject (S-H), the tail of the subject (S-T), the head of the object (O-H), the tail of the object (O-T), and the nonentity words (O). For the entity in each triplet, we only tag the head and tail of the entity and tag the other parts of the entity with O. For relation information, we divide the other four regions except the O region into labels for the number of relations. In other words, the length of the label corresponding to each word is $4N + 1$, where N is the number of preset relations. We are also inspired by the binary tagging method [8]. When the network recognizes the subject of a relation, we tag the position of the corresponding relations in the S-H and S-T region as 1, and when the object is recognized, we make the same label in the O-H and O-T regions. Since each entity can be in multiple relational triplets or both subject and object, there may be multiple labels with a value of 1 of each word, like Leonardo is in both (Leonardo DiCaprio, Act, Jack) and (Leonardo DiCaprio, Work_in, Titanic) (see Figure 2). In other words, the task of tagging each word is a multi-classification task. This tagging approach can be a good solution to EPO and SEO problems.

3.2. Relational Triplet Judgment Network. In order to solve the RO problem, we propose a relational triplet judgment network (see Figure 3). First, we extract the subject and object words encoded by the pretrained model based on the results of the joint entity and relationship labels. Then, we use relative position concerns to embed the words of the subject and object into the relative position information. We combine the subject and the object in pairs and add the sentence information to determine whether the subject and object are a triplet.

3.2.1. BERT Encoder. The encoder can extract features from the sentence information S_i and convert S_i into a word embedding X_i . The output word embedding X_i can be used for the prediction and tagging of subsequent modules. We use BERT [21, 22] encoder to extract features from sentence information.

Here we briefly review the overview of BERT. BERT is a language representation model composed of a multilayer bidirectional transformer [23] encoder. Through self-supervised training on a large number of unmarked corpora, the BERT model contains rich language knowledge. In the training process, BERT learns the deep representation of words by randomly masking or replacing some words and predicting through context and has achieved surprising results in multiple tasks. We denote the BERT model as $B(x)$ and denote each layer of the BERT model as $T(x)$. The operation process of the BERT can be expressed as

$$H = B(E_t + E_p + E_s), \quad (1)$$

and the operation of each layer of BERT can be expressed as

$$\begin{aligned} H^0 &= T(WO_t + E_p + E_s), \\ H^l &= T(H^{l-1}), \end{aligned} \quad (2)$$

where E_t is word split embedding, E_p is position embedding, E_s is sentence segmentation embedding, O_t is one-hot embedding of input words, and W is the one-hot embedding stored by BERT.

3.2.2. Relative Positional Attention. The relative position is important information for judging whether the subject and the object are in the same triplet. Generally, the length of the relative position distance between entities in the same relational triplet is shorter than the distance of the relative position with other entities. But not all subjects and objects in the same triplet are close. In order to solve this problem, inspired by relative position representations in machine translation [24], we add an attention layer with learnable relative positional embedding in the output layer of the encoder.

On the basis of the attention mechanism, we add the relative position information between the subject and the object. Specifically, in order to obtain more comprehensive position information, we use two vectors to represent the relative positional encoding between each word and learn the relative positional information of the entity character

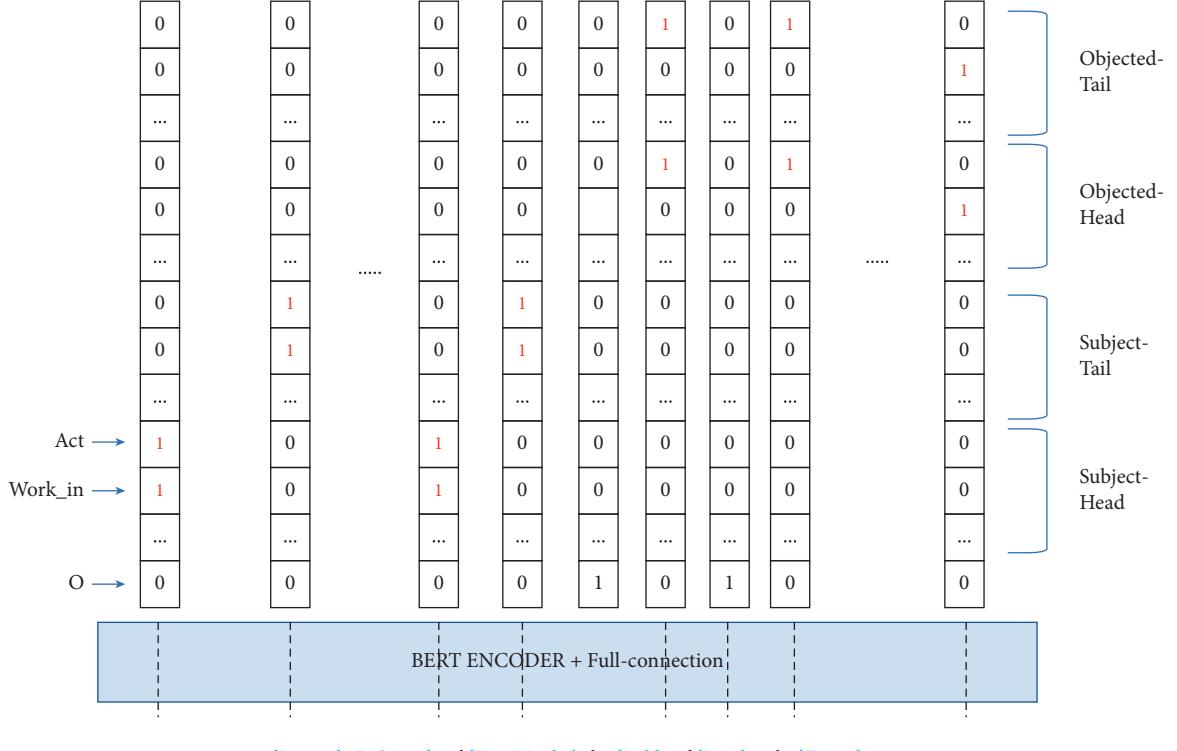
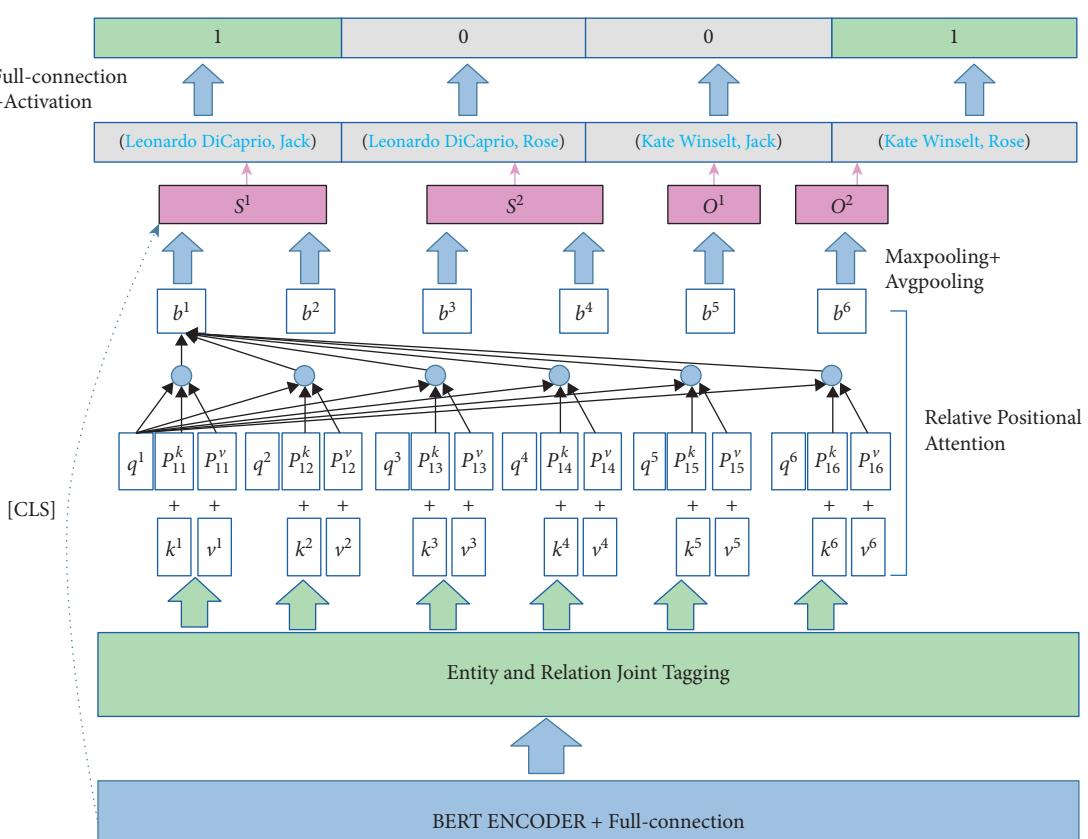


FIGURE 2: The joint tagging method of entities and relations. The label of each word is divided into five regions, and the other four regions except the O region are divided into the number of relation label.



[Leonardo DiCaprio] and [Kate Winslet] play [Jack] and [Rose] in the [Titanic]

FIGURE 3: Overview of triplet judgment network with entity and relation joint tag. In this example, two relational triplets have a relation overlapping (RO) problem.

level while calculating the subject and object attention. We use the vectors P_{so}^k and P_{so}^v to represent the relative positional information between the subject S and the object O. The relative positional vectors P_{so}^k and P_{so}^v add to the calculation process of the key and value in the attention, respectively. The detailed operation is as follows:

$$\begin{aligned}\alpha_{so} &= \alpha_s W^q \cdot (\alpha_s W^k + P_{so}^k), \\ \alpha'_{so} &= \text{Softmax}(\alpha_{so}), \\ b_s &= \sum_{o=1}^n \alpha'_{so} (a_o W^v + P_{so}^v),\end{aligned}\tag{3}$$

where W^q , W^k , and W^v represent the weight matrix of query, key, and value in the attention mechanism, respectively. α_s and a_s represent the word embedding in the subject entity and the object entity, respectively. We set a limit for the relative position; that is, when the relative distance of the word exceeds the set maximum length, we treat it as the maximum distance. The maximum distance we set is 50. The detailed operation is as follows:

$$\begin{aligned}P_{so}^k &= W_{\text{len}(o-s, L_{\max})}^k, \\ P_{so}^v &= W_{\text{len}(o-s, L_{\max})}^v, \\ \text{len}(x, L_{\max}) &= \max(L_{\max}, \text{abs}(x)),\end{aligned}\tag{4}$$

where L_{\max} is our maximum distance.

3.2.3. Entity Feature Extraction. Through the joint tagging of entities and relations and relative positional attention, we can get all the subjects and objects in the sentence with relative position information and relations information. In order to extract the feature of the entities as comprehensively as possible, we extract each subject and object through two processes of average pooling and maximum pooling; and, for the entity embedding to fuse the global feature of the sentence instead of the feature of the surrounding words, we concatenate the [CLS] from the BERT as sentence embedding into the entity embedding. Then, we enumerate all possible combinations of subject and object in the same relation and judge whether it is the correct combination. The specific operation is as follows:

$$\begin{aligned}S &= \text{Concat}\left(\text{MaxPooling}\left(\sum_{i=1}^s S_i\right), \text{AvgPooling}\left(\sum_{i=1}^s S_i\right), E_{\text{cls}}\right), \\ O &= \text{Concat}\left(\text{MaxPooling}\left(\sum_{j=1}^o O_j\right), \text{AvgPooling}\left(\sum_{j=1}^o O_j\right), E_{\text{cls}}\right), \\ E_{so} &= \text{Concat}(S, O), \\ E'_{so} &= \text{Sigmoid}(WE_{so} + b),\end{aligned}\tag{5}$$

where S and O are subject and object, respectively. S_i and O_j are word embedding in a subject and an object, respectively. E_{cls} is the output vector at [CLS] in BERT.

3.3. Loss Function. Our model is divided into two stages. In the first stage, we use BECWithLogits Loss to learn the joint tagging of entities and relations. In order to reduce the influence of sparse tags on model learning, we square the probability value output by the model to make the result smoother. In the second stage, the relational triplet judgment network uses CrossEntropy Loss. The losses of these two stages are added together in a certain proportion and jointly trained.

$$\begin{aligned}\text{Loss}_{\text{CE}} &= -[y_n * \log(x_n) + (1 - y_n) * \log(1 - x_n)], \\ \alpha &= \begin{cases} 1, & x_n < 0, \\ 2, & x_n \geq 0, \end{cases} \\ x_{n\text{-smooth}} &= (-1)^\alpha * (x_n)^2, \\ \text{Loss}_{\text{BCE}} &= -[y_n * \log(\sigma(x_{n\text{-smooth}})) + (1 - y_n) * \log(1 - \sigma(x_{n\text{-smooth}}))], \\ \text{Loss} &= \text{Loss}_{\text{BCE}} + \lambda * \text{Loss}_{\text{CE}},\end{aligned}\tag{6}$$

where x_n is the output result of the n th batch size, y_n is the label of the n th batch size, and λ is the ratio of the addition of the loss functions; we set it to 0.001.

4. Experiments

In this work, we designed three experiments to evaluate our network. First, we used the entire test set to evaluate the performance of the model to reflect the model's ability to deal with common problems. Then, in order to evaluate the performance of the model in processing complex sentences, we tested the model in test sets with overlapping problems and different numbers of triples.

4.1. Datasets. For comparison with the previous models in this field, we select one of the most popular datasets: NYT. For proving our model has good performance in multiple languages, we also evaluate our model on Chinese datasets: DuIE 2.0 and CMED. For showing the differences between the three datasets, we analyze and compare them (see Table 1). NYT is a dataset in the English news field and it contains 25 kinds of relations. DuIE 2.0 is a Chinese general field dataset containing 54 kinds of relations. CMED is a Chinese medical field dataset containing 43 kinds of relations. We also analyze the overlap problem and the number of triples in each dataset. These three datasets have different characteristics that can comprehensively evaluate the performance of the model.

4.2. Implementation Details. Our model is implemented using PyTorch and the optimizer of the network framework is AdamW [25]. In terms of model parameters, the batch size in all our experiments is 20, the learning rate is reduced from 0 to 0.0003 and then to 0 as the number of training increases, the maximum sequence length is 256, and the word out of range is ignored. In order to obtain more accurate information in the second stage of the model, we train separately

TABLE 1: Statistic of the datasets.

Dataset	Train.	Valid.	Test.	Overlapping pattern			Number of triplets				
				SEO	EPO	RO	$N=1$	$N=2$	$N=3$	$N=4$	$N \geq 5$
NYT	56195	5000	5000	1297	978	690	3244	1045	312	291	108
DuIE 2.0	173108	20674	50583	8009	2652	3734	12242	4488	1603	1019	1322
CMED	14339	3585	4482	2161	67	1805	1380	779	433	312	681

for 10 epochs in the first stage and then train jointly. We use Tesla V100 to train our model for up to 50 epochs and evaluate the model on the validation set. We select the best model and output the final result on the test set. When training on the NYT dataset, the pretrained model we use is BERT-base. When training on the DuIE 2.0 and CMED datasets, the pretrained model we use is BERT-wwm-ext.

4.3. Baselines and Evaluation Metrics. We select the advanced models in this field in recent years for comparison. (1) NovelTagging [16] proposed the joint tagging method of entities and relations for the first time, but it did not solve the overlapping problems. (2) CopyRE [11] first proposed using the encoder-decoder structure for the extraction of entities and relations tasks. (3) GraphRel [17] used graph convolutional networks to combine the features of all words to optimize the performance of the model. (4) ETL-Span [19] added the strategy with span information to the extraction model. (5) WDec [12] used seq2seq to generate word sequences. (6) CopyMTL [26] used multitask learning framework to combine extraction of entities and relations. (7) RSAN [13] proposed a relation-specific attention network to solve the overlapping problems. (8) CasRel [8] put forward the concept that the relations are a function of subject to object.

When the triplet output by the model is exactly the same as the label, we judge it as the correct result. We use Precision (Prec), Recall (Rec), and F1 score as indicators of the evaluation model.

4.4. Experimental Results and Analysis

4.4.1. Main Results. From the results shown in Table 2, we can find that our model on the NYT dataset surpassed all baseline models in the comprehensive index F1 score; and Table 3 shows the evaluation results of our model and the best baseline model in the Chinese datasets DuIE 2.0 and CMED; our model is still the best in F1 score. The test results prove that the comprehensive performance of our model is better than those of all baseline models. Compared with the best baseline model, the F1 value of our model is improved by 1.1 on the NYT dataset, 6.0 on the DuIE 2.0 dataset, and 5.6 on the CMED dataset; and these three datasets of experiments also prove that the performance of our model is not limited to a single language and it gets the best results under multiple language tests.

We analyze the reasons why RTJTN can get better performance and summarized the following points: (1) Although the SOTA model CasRel has achieved exciting results, it is essentially a two-stage model. The error of

TABLE 2: Results on the NYT dataset.

Models	Prec	Rec	F1
NovelTagging	62.4	31.7	42.0
CopyRE	61.0	56.6	58.7
GraphRel	63.9	60.0	61.9
CopyMTL	75.7	68.7	72.0
ETL-Span	84.9	72.3	78.1
WDec	94.5	76.2	84.4
RSAN	85.7	83.6	84.6
CasRel	89.7	89.5	89.6
RTJTN	92.5	89.0	90.7

TABLE 3: Results on the DuIE 2.0 and CMED datasets.

Models	DuIE 2.0			CMED		
	Prec	Rec	F1	Prec	Rec	F1
CasRel	70.2	70.0	70.1	56.8	46.7	51.3
RTJTN	76.6	75.6	76.1	61.8	51.8	56.4

extracting entities directly affects the result of extracting relations. Our model combines entity extraction and relation extraction into one stage to eliminate error iteration. (2) The CasRel model does not make obvious feature distinctions for the relational triples in the RO problem but only achieves the goal through the self-learning of the model. Compared with our model, in order to distinguish the features between the relational triples, attention information with relative positions is added. (3) The performance improvement of our model on the DuIE 2.0 dataset and the CMED dataset is much higher than that of the NYT dataset. Our analysis found that the number of relations between the DuIE 2.0 dataset and the CMED dataset far exceeds that of NYT and the average sentence length is also much longer than the sentences in the NYT dataset. We believe that the more there is triple information in the sentence, the greater is the error iteration that CasRel suffers. On the contrary, our model does not have this shortcoming.

4.4.2. Ablation Study of RTJTN. In order to evaluate the contribution of each part of the model to the results, we performed an ablation study on the NYT dataset. Starting from the complete model, we remove part of the structure of the model every time and observe the effect of this structure on the result, which is shown in Table 4. (1) Relation Positional Attention can effectively provide relative position information between subject and object. (2) Information of Sentence enables word embedding information to get richer semantic information. (3) We replaced the original

TABLE 4: Ablation study on NYT dataset.

Models	Prec	Rec	F1
RTJTN	92.5	89.0	90.7
Relation Positional Attention	91.7	89.0	90.5
Information of Sentence	91.4	89.1	90.2
Maxpooling and Average Pooling	91.2	88.5	89.8
Triplet Judgment Network	87.1	89.5	88.3

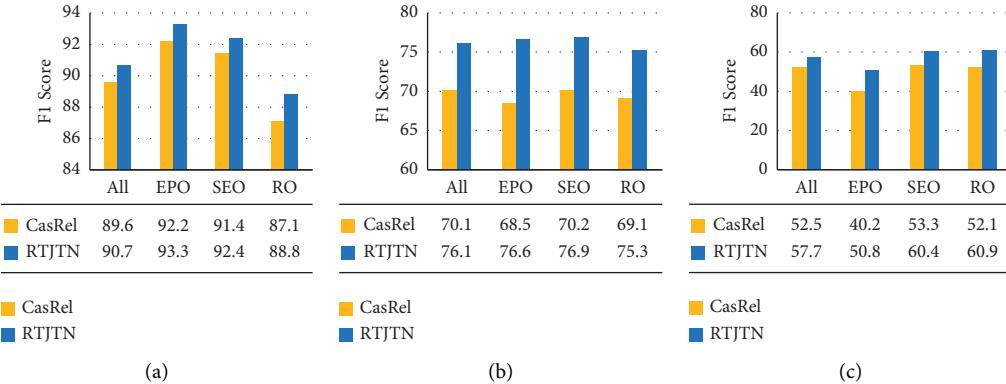


FIGURE 4: Relation extraction on sentences with overlapping problems. (a) F1 on the NYT dataset. (b) F1 on the DuIE 2.0 dataset. (c) F1 on the CMED dataset.

TABLE 5: F1 score of sentences with different number of triplets.

Models	Number of triplets														
	NYT					DuIE 2.0					CMED				
	N = 1	N = 2	N = 3	N = 4	N ≥ 5	N = 1	N = 2	N = 3	N = 4	N ≥ 5	N = 1	N = 2	N = 3	N = 4	N ≥ 5
CasRel	88.2	90.3	91.9	94.2	83.7	68.5	66.9	68.3	70.2	72.1	46.0	45.2	47.9	49.5	54.6
RTJTN	88.2	90.6	91.2	95.1	90.3	76.1	74.1	74.6	77.0	78.2	44.4	49.3	56.3	58.4	64.6

Maxpooling and Average Pooling with the word embedding in the entity directly splicing, and the F1 score dropped significantly. (4) We remove the relational triplet judgment network and only keep the F1 score of the joint entities and relations tagging layer, which shows that our network is of great help in solving the RO problem. After joining relational triplet judgment network training, the F1 score of our joint entities and relations tagging also increased from 88.28 to 83.34, indicating that the relational triplet judgment network is not only helpful to the final result but also helpful to the training of the joint tagging layer.

4.4.3. Analysis on Overlapping Cases. The overlapping problems of entities and relations impact traditional tagging methods and have a great impact on the final result. In order to prove that our model can effectively solve the overlapping problems, we separately evaluate our model on three datasets of the overlapping problems. We divide the overlapping problems into three types, EntityPairOverlap (EPO), SingleEntityOverlap (SEO), and RelationOverlap (RO), and extract the data of these three situations from NYT, DuIE 2.0, and CMED datasets. We compare our model with the best baseline model in three cases and the results are shown

in Figure 4. From the results in the figure, we can see that our model surpasses the baseline model in all three overlapping problems.

When testing on the RO problem datasets, the performance of the baseline model is worse than the result of testing on the complete data set, which also proves that the RO problem can have a negative impact on the performance of the model. The results in Figure 4 show that when the model predicts sentences with RO problems, the negative impact of our model is significantly less than the best baseline model. This proves that our model can better predict the relational triplets with RO problems.

4.4.4. Analysis on Different Numbers of Relational Triplets. The number of relational triplets in the text also has a huge impact on the results of the extraction task. Generally, the more the number of relational triplets is in a sentence, the more difficult it is to extract the correct relational triplets. In order to evaluate the performance of our model for extracting different numbers of triplets in sentences, we divide the sentences in the NYT, DuIE 2.0, and CMED datasets into five categories, which represent sentences containing 1, 2, 3, 4, and ≥5 triplets in sentences. The results

are shown in Table 5. From the results in the table, we can find that when the sentence contains a small number of triplets, the performances of the two models are not much different, but when the number of triplets increases, the results of our model are much better than the baseline model. Therefore, our model is better at extracting complex triplets than other models.

5. Conclusions

In this paper, we propose a relational triplet joint tagging network (RTJTN) which contains a joint tagging of entities and relations and a relational triplet judgment network. Instead of extracting entities and relations in unstructured sentences separately, our tagging method completely combines entity extraction and relation extraction into one task and effectively solves the problems of SPO and EPO. As a consequence, our model can effectively avoid error iteration and get better performance than the baseline model. In addition, we use the RO problem as an indicator of the evaluation model for the first time and our relational triplet judgment layer has achieved excellent results on the RO problem. In the evaluation of NYT, DuIE 2.0, and CMED datasets, our model has also made significant improvements in performance. The results of our model are also better than those of the baseline model in experiments with various overlapping problems and the different number of triplets problems.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

- [1] D. Nadeau and S. Sekine, “A survey of named entity recognition and classification,” *Lingvisticae Investigationes. International Journal of Linguistics and Language Resources*, vol. 30, no. 1, pp. 3–26, 2007.
- [2] D. Zelenko, C. Aone, and A. Richardson, “Kernel methods for relation extraction,” *Journal of Machine Learning Research*, vol. 3, pp. 1083–1106, 2003.
- [3] X. Yu and W. Lam, “Jointly identifying entities and extracting relations in encyclopedia text via a graphical model approach,” in *Proceedings of the Coling 2010: Posters*, pp. 1399–1407, Beijing, China, August 2010.
- [4] Q. Li and H. Ji, “Incremental joint extraction of entity mentions and relations,” in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pp. 402–412, Baltimore, MD, USA, June 2014.
- [5] X. Ren, Z. Wu, W. He et al., “Cotype: joint extraction of typed entities and relations with knowledge bases,” in *Proceedings of the 26th International Conference on World Wide Web*, pp. 1015–1024, Perth Australia, April 2017.
- [6] P. Gupta, H. Schütze, and B. Andrassy, “Table filling multi-task recurrent neural network for joint entity and relation extraction,” in *Proceedings of the COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 2537–2547, Osaka, Japan, December 2016.
- [7] A. Katiyar and C. Cardie, “Going out on a limb: joint extraction of entity mentions and relations without dependency trees,” in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 917–928, Vancouver, Canada, July 2017.
- [8] Z. Wei, J. Su, Y. Wang, Y. Tian, and Y. Chang, “A novel cascade binary tagging framework for relational triple extraction,” 2019, <https://arxiv.org/abs/1909.03227>.
- [9] S. Wu and Y. He, “Enriching pre-trained language model with entity information for relation classification,” in *Proceedings of the 28th ACM international conference on information and knowledge management*, pp. 2361–2364, Beijing, China, November 2019.
- [10] Y. Zhang and J. Yang, “Chinese NER using lattice LSTM,” 2018, <https://arxiv.org/abs/1805.02023>.
- [11] X. Zeng, D. Zeng, S. He, K. Liu, and J. Zhao, “Extracting relational facts by an end-to-end neural model with copy mechanism,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 506–514, Melbourne, Australia, July 2018.
- [12] T. Nayak and H. T. Ng, “Effective modeling of encoder-decoder architecture for joint entity and relation extraction,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 8528–8535, Hilton New York Midtown, NY, USA, February 2020.
- [13] Y. Yuan, X. Zhou, S. Pan, Q. Zhu, Z. Song, and L. Guo, “A relation-specific attention network for joint entity and relation extraction,” in *Proceedings of the International Joint Conference on Artificial Intelligence 2020*, pp. 4054–4060, Yokohama, Japan, November 2020.
- [14] S. Riedel, L. Yao, and A. McCallum, “Modeling relations and their mentions without labeled text,” in *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 148–163, Barcelona, Spain, September 2010.
- [15] M. Miwa and Y. Sasaki, “Modeling joint entity and relation extraction with table representation,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1858–1869, Doha, Qatar, October 2014.
- [16] S. Zheng, F. Wang, H. Bao, Y. Hao, P. Zhou, and B. Xu, “Joint extraction of entities and relations based on a novel tagging scheme,” 2017, <https://arxiv.org/abs/1706.05075>.
- [17] T.-J. Fu, P.-H. Li, and W.-Y. Ma, “GraphRel: modeling text as relational graphs for joint entity and relation extraction,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1409–1418, Florence, Italy, July 2019.
- [18] G. Bekoulis, J. Deleu, T. Demeester, and C. Develder, “Joint entity recognition and relation extraction as a multi-head selection problem,” *Expert Systems with Applications*, vol. 114, pp. 34–45, 2018.
- [19] B. Yu, Z. Zhang, X. Shu et al., “Joint extraction of entities and relations based on a novel decomposition strategy,” 2019, <https://arxiv.org/abs/1909.04273>.
- [20] J. Liu, S. Chen, B. Wang, J. Zhang, N. Li, and T. Xu, “Attention as relation: learning supervised multi-head self-attention for relation extraction,” in *Proceedings of the Twenty-Ninth*

International Joint Conference on Artificial Intelligence,
Yokohama, Japan, July 2020.

- [21] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: pre-training of deep bidirectional transformers for language understanding,” 2018, <https://arxiv.org/abs/1810.04805>.
- [22] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving language understanding by generative pre-training,” 2018, <https://s3-us-west-2.amazonaws.com/openai-assets/research-h-covers/language-unsupervised/languageunderstandingpaper.pdf>.
- [23] A. Vaswani, N. Shazeer, N. Parmar et al., “Attention is all you need,” 2017, <https://arxiv.org/abs/1706.03762>.
- [24] P. Shaw, J. Uszkoreit, and A. Vaswani, “Self-attention with relative position representations,” 2018, <https://arxiv.org/abs/1803.02155>.
- [25] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” 2017, <https://arxiv.org/abs/1711.05101>.
- [26] D. Zeng, H. Zhang, and Q. Liu, “Copymtl: copy mechanism for joint extraction of entities and relations with multi-task learning,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 9507–9514, Hilton New York Midtown, NY, USA, February 2020.