

## Article

# Integration of Relation Filtering and Multi-Task Learning in GlobalPointer for Entity and Relation Extraction

Bin Liu <sup>1,2</sup> , Jialin Tao <sup>1,2</sup>, Wanyuan Chen <sup>1,2</sup> , Yijie Zhang <sup>1,2</sup>, Min Chen <sup>1,2</sup>, Lei He <sup>1,2,\*</sup>  and Dan Tang <sup>1,2</sup><sup>1</sup> School of Software Engineering, Chengdu University of Information Technology, Chengdu 610225, China<sup>2</sup> Sichuan Province Engineering Technology Research Center of Support Software of Informatization Application, Chengdu 610225, China

\* Correspondence: helei1978@cuit.edu.cn

**Abstract:** The rise of knowledge graphs has been instrumental in advancing artificial intelligence (AI) research. Extracting entity and relation triples from unstructured text is crucial for the construction of knowledge graphs. However, Chinese text has a complex grammatical structure, which may lead to the problem of overlapping entities. Previous pipeline models have struggled to address such overlap problems effectively, while joint models require entity annotations for each predefined relation in the set, which results in redundant relations. In addition, the traditional models often lead to task imbalance by overlooking the differences between tasks. To tackle these challenges, this research proposes a global pointer network based on relation prediction and loss function improvement (GPRL) for joint extraction of entities and relations. Experimental evaluations on the publicly available Chinese datasets DuIE2.0 and CMeIE demonstrate that the GPRL model achieves a 1.2–26.1% improvement in F1 score compared with baseline models. Further, experiments of overlapping classification conducted on CMeIE have also verified the effectiveness of overlapping triad extraction and ablation experiments. The model is helpful in identifying entities and relations accurately and can reduce redundancy by leveraging relation filtering and the global pointer network. In addition, the incorporation of a multi-task learning framework balances the loss functions of multiple tasks and enhances task interactions.

**Keywords:** knowledge graph; joint entity and relation extraction; overlapping triplets; multi-task learning



**Citation:** Liu, B.; Tao, J.; Chen, W.; Zhang, Y.; Chen, M.; He, L.; Tang, D. Integration of Relation Filtering and Multi-Task Learning in GlobalPointer for Entity and Relation Extraction. *Appl. Sci.* **2024**, *14*, 6832. <https://doi.org/10.3390/app14156832>

Academic Editor: Fabrizio Marozzo

Received: 20 June 2024

Revised: 24 July 2024

Accepted: 3 August 2024

Published: 5 August 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Currently, knowledge presents explosive growth in the world, and people pay more attention to seeking valuable information online. In the data-driven world, the primary challenge is learning and extracting valuable information from text. Knowledge graphs have been a popular tool in many research fields, helping individuals to access knowledge more intuitively [1]. Now, knowledge graphs are widely used in healthcare [2], financial services [3], intelligent transport, and other fields [4]. However, most of the information is unstructured or semi-structured data, for which the direct understanding and construction of knowledge graphs by computer systems are not possible [5]. So, natural language processing (NLP) is applied to extract valuable structured information from large amounts of unstructured or semi-structured text [6]. Entity recognition and relation extraction are two crucial subtasks of information extraction in NLP; they have been popular in identifying triplets from unstructured text in the form of <subject, relation, object> [7].

During the triplet extraction process, complex relationships within a single sentence lead to the phenomenon of multiple triplets in the same subject, object, or relation, as well as nested entities. According to the degree of entity overlap, the problem of triple overlap is divided into three categories: normal, entity pair overlap (EPO), and single-entity overlap (SEO) [8]. Entity pair overlap indicates that multiple relations are present between a pair of entities, while single-entity overlap refers to multiple relationships and entities

sharing the same entity. Many researchers choose to operate within a single sentence and assume only one entity pair per sentence, which struggles to identify overlapping triplets and nested entities effectively. Focused on the above problems of overlapping triples and nested entities, several models have been proposed to address these two issues. However, a lot of models often require entity annotations for each predefined relation in the set, which leads to redundant relation judgments. In addition, the multi-task problem also remains during the joint extraction of entities and relations. Optimizing the loss function is a key strategy in multi-task learning; many entity relation extraction models neglect the differences between input and output among subtasks, which may lead to domination by a certain task and a significant reduction in the impact of losses from other tasks. A common approach is to manually set weights, which can result in extensive experimentation and development costs.

This research proposes a global pointer network based on relation prediction and loss function improvement (GPRL) to address the issues mentioned above. The GPRL involves relation filtering and adaptive loss weight adjustment. The method of relation filtering is used to solve the problem of redundant relation judgment, which includes predicting relations first, filtering out impossible redundant relations, reducing computational costs, and extracting entity relation triplets. The model has two subtasks: relation filtering and joint extraction of entities and relations, which employs a multi-task learning strategy to learn multiple tasks simultaneously through shared models. Fixed weights can also affect the model's performance because of different convergence speeds for various tasks. Therefore, this research adopted an adaptive loss weight adjustment strategy to balance the loss functions of multiple tasks, which can dynamically adjust performance based on learning speeds, difficulty levels, and effectiveness of learning.

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

1. A global pointer decoding approach is introduced in this research to address the issues of overlapping triplets and nested entities. Entity and relation information is encapsulated in a dual-head matrix, which effectively resolves the problems.
2. The global pointer-based model is expanded to implement two enhancements. Firstly, relation filtering is applied to combat redundant relationships, leading to a notable reduction in the evaluation of such redundancies. Secondly, a dynamic loss balancing strategy is applied to solve the complicated problems of multi-task learning, and the team has embraced it, encouraging improved interactions among various tasks.

## 2. Related Work

The extraction of entities and relations based on deep learning is mainly divided into pipeline and joint extraction methods [9]. Previous pipeline methods primarily used two major classes of structures to separate entity and relation extraction: Convolutional Neural Networks (CNNs) [10] and Recurrent Neural Networks (RNNs) [11]. Subsequent research based on CNNs and RNNs led to many variants such as Long Short-Term Memory (LSTM) [12] and Bidirectional Long Short-Term Memory (Bi-LSTM) [13]. Xu et al. developed an SDP-LSTM relation extraction model based on the shortest path of the syntactic dependency parsing tree and improved its accuracy by incorporating word vector features, part-of-speech features, WordNet features, and syntactic type features [14]. Zhang et al. utilized the Bi-LSTM model to combine information before and after the current word for relation extraction [15]. Ling et al. introduced an attention mechanism to Bi-LSTM for capturing both local and global features of input sentences [16]. Although the pipeline method has achieved more success in entity relation extraction, its splitting approach often overlooks the connections between entity relations, which leads to issues such as interaction omissions, error propagation, and entity redundancy. The joint extraction method involves extracting entities and relations simultaneously, which can leverage the interaction information between entities and relations to enhance the model's performance. Miwa et al. developed an end-to-end entity relation extraction method to share parameters between entity recognition and relation extraction tasks to strengthen the correlation between such

two tasks [17]. The above models only shared the bidirectional sequence LSTM representation at the encoding layer and only treated named entity recognition and relation extraction as two separate tasks rather than performing true joint extraction. However, the introduction of these models still laid the foundation for subsequent joint extraction methods. Zheng et al. proposed a joint tagging strategy called NovelTagging, which achieved joint decoding. They transformed the entity relation extraction problem into a sequence labeling problem, enabling the simultaneous identification of entities and relations [18].

Focused on the problems of overlapping triples and nested entities, several models have been proposed to address these two issues. For example, Yu et al. proposed a new strategy called ETL-Span, which decomposed the joint extraction task into two tasks based on Span's labeling scheme. The two tasks applied are divided into head entity extraction and tail entity relation extraction [19]. Based on ETL-Span, Wei et al. proposed a cascaded binary annotation framework known as CasRel, which first identified all possible subjects in a sentence by using a start position classifier and an end position classifier. Further, for each subject, span-based labeling is applied to identify the corresponding objects with a classifier based on each relation [20]. Several researchers have also improved relative methods by introducing sentence encoding with the general framework of this model, with examples including IDCNN (Iterated Dilated CNN) and Bi-LSTM [21]. The CB-Chinese-Bert integrated self-attention mechanism for Chinese text encoding was presented, and it can be effective in obtaining richer semantic features in sentences [22]. In addition, though Casrel addresses the overlap issues, its framework still suffers from low computational efficiency and numerous requirements for evaluating redundant relationships. Moreover, the framework identification of nested entities requires more accurate results. Wang et al. proposed a model called TPLinker designed to extract information in a single stage. The model unifies the extraction tagging framework into a character pair linking problem, which allows for multi-head tagging through matrices and aligns the subject with the object under specific relations [23]. Subsequently, based on TPLinker and CasRel, Zhang et al. proposed a triple joint extraction framework called PRGC, which is based on latent relationships and global communication. However, this method still needs to improve the generalization ability of overlapping entities. Additionally, the structure of PRGC is complex and requires high training processes [24]. Su et al. proposed a global normalization approach called GlobalPointer for identifying both nested and non-nested entities [25]. The approach involves constructing an entity matrix with a tagging method similar to TPLinker and training the model by incorporating a new multi-label cross-entropy loss function. Based on GlobalPointer, researchers proposed a joint entity–relation extraction model called GPLinker, which utilizes multiple GlobalPointers for extracting entities and relationships [26]. Regular multiple pointer networks have been applied to recognize multiple entity relationship categories. Each pointer network introduced two modules to identify the head and tail of the entity [19,20]. During training and evaluation, a single module can only consider the information of either the entity's head or tail, which results in inconsistencies in taking the global information of the entity into account during training and prediction processes. However, the global pointer network treats the head and tail positions of the entity simultaneously, which can supply the global perspective for the model. So, consistency in training and prediction goals can be achieved, and the performance of the entity relationship extraction model can be enhanced [27]. TPLinker also introduced the multi-label cross-entropy loss function from GPLinker and TPLinker-plus to solve category imbalance problems [23]. However, the discussed models often require entity annotations for each predefined relation in the set, which leads to redundant relation judgments. In addition, the multi-task problem also remained during the joint extraction of entities and relations.

Multi-task learning (MTL) is a powerful machine learning algorithm that leverages the inherent relationships between multiple related tasks to enhance overall model performance [28]. Further, MTL can be valuable in overcoming the challenges associated with data scarcity, which can alleviate model overfitting and significantly enhance model performance. MTL generally employs parameter sharing for learning, which can pro-

vide data augmentation in scenarios for multiple tasks that share commonalities. Such approaches can enhance the model's performance compared to training separate models on individual datasets [29]. With a focus on language processing, MTL is widely employed, especially for the main tasks of entity relation extraction and relation filtering. The varying importance and data distributions of different tasks often lead to imbalance when directly summing the loss functions from all tasks in most models. So, more research has been focused on strategies and optimization methods for adjusting loss function weights in MTL to improve model performance and training effectiveness. The advanced methods include Gradient Normalization (GradNorm), Dynamic Weight Averaging (DWA), Dynamic Task Prioritization (DTP), and Uncertainty to Weigh Losses (UWL). GradNorm was defined to equalize the gradient magnitude of different tasks, aiming to ensure similar learning speeds across tasks [30]. DWA, like GradNorm, aimed to balance learning speeds among tasks but focused on adjusting weights based on the change ratio of each task's loss to assess its difficulty. If tasks with a faster decrease in loss were easier to learn, the higher weights assigned to tasks presented slower learning speeds [31]. DTP introduced the concept of task priority by categorizing tasks based on difficulty, adaptively adjusting weights, and assigning higher weights to harder tasks during each training session [32]. UWL introduced noise as a measure of task difficulty, assigning higher weights to tasks with lower noise levels and higher certainty [33].

### 3. Materials and Methods

The GPRL model proposed in this research is applied for extracting entities and relationships with a global pointer network, which is integrated with relation filtering and multi-task learning methods. Figure 1 depicts the model structure consisting of three modules: the coding, the global pointer network decoding, and the relation filtering decoding. Firstly, the set  $\{x_1, x_2, \dots, x_{11}\}$  represents a segment of text, where  $x_i$  denotes each word in the text. These words are processed through the Chinese-BERT-WWM encoding module for pre-training, converting the text into word vectors containing contextual information, i.e.,  $(h_0, h_1, h_2, \dots, h_n)$ . Secondly, the global pointer module skillfully extracts entities from the word vectors in the form of a matrix. Here,  $S(sh, st)$  represents the extraction of the head and tail of the subject, and  $S(oh, ot)$  represents the extraction of the head and tail of the object. Figure 1 shows that the subject  $(x_1, x_2)$  and the object  $(x_4, x_6)$  have been extracted. Simultaneously, the relation filter extracts potential relationships from the word vectors through a pooling layer, a linear layer, and a Sigmoid activation function. Further,  $r_1$  is extracted as the potential relationship for this sentence, and all potential relationships are placed in the set  $\{r_1, r_2, \dots, r_i\}$ . Finally, by using the global pointer network and combining the extracted subjects, objects, and potential relationships, the final triplet information can be extracted. For example, the final triplet information  $(x_1x_2, r_1, x_4x_5x_6)$  is extracted through the head of the subject, the head of the object, and the relation  $S(sh, oh | r)$ , as well as the tail of the subject, the tail of the object, and the relation  $S(st, ot | r)$ .

#### 3.1. Coding Module

Chinese-BERT-WWM is a pre-trained model [34] for the Chinese language based on BERT (Bidirectional Encoder Representations from Transformers) [35]. Machine learning models are taken as pre-trained models in research on large datasets in advance, and they can convert text files into word vectors to obtain a general language representation. Chinese-Bert-WWM has been specifically trained on a large corpus of Chinese text. One of its unique features is the "Whole Word Masking" (WWM) technique used in the pre-training process. Compared with the traditional BERT model, it can mask individual characters, and an entire word masked in Chinese-BERT-WWM can better capture word-level semantic information. The process is as follows: Firstly, sentences are converted into word vector embeddings. The Chinese-Bert-WWM model can learn the vector representations of each word through contextual semantic information. Compared with traditional pre-trained language models such as One-hot [36] and Word2Vec [37], the Chinese-Bert-WWM model

takes advantage of the consideration of the word position vector. Since a word may express different meanings in different positions, the position information of words is not ignored during the entire word embedding process, which can solve the problem of polysemy. The input sequence  $S = [w_1, w_2, \dots, w_n]$  of the Chinese-Bert-WWM model maps  $S$ , where the sequence length is  $n$ . The beginning and end of the sequence are marked with “[CLS]” and “[SEP]”, respectively, so the text vector representation output after Chinese-Bert-WWM encoding is as follows:

$$V = ([CLS], w_1, w_2, \dots, w_n, [SEP]) = (h_1, h_2, \dots, h_n) \tag{1}$$

where  $V$  represents the output text vector result after encoding,  $w_i$  represents the  $i$ th character of the input sequence, and  $h_i$  represents the corresponding embedding vector of each character.

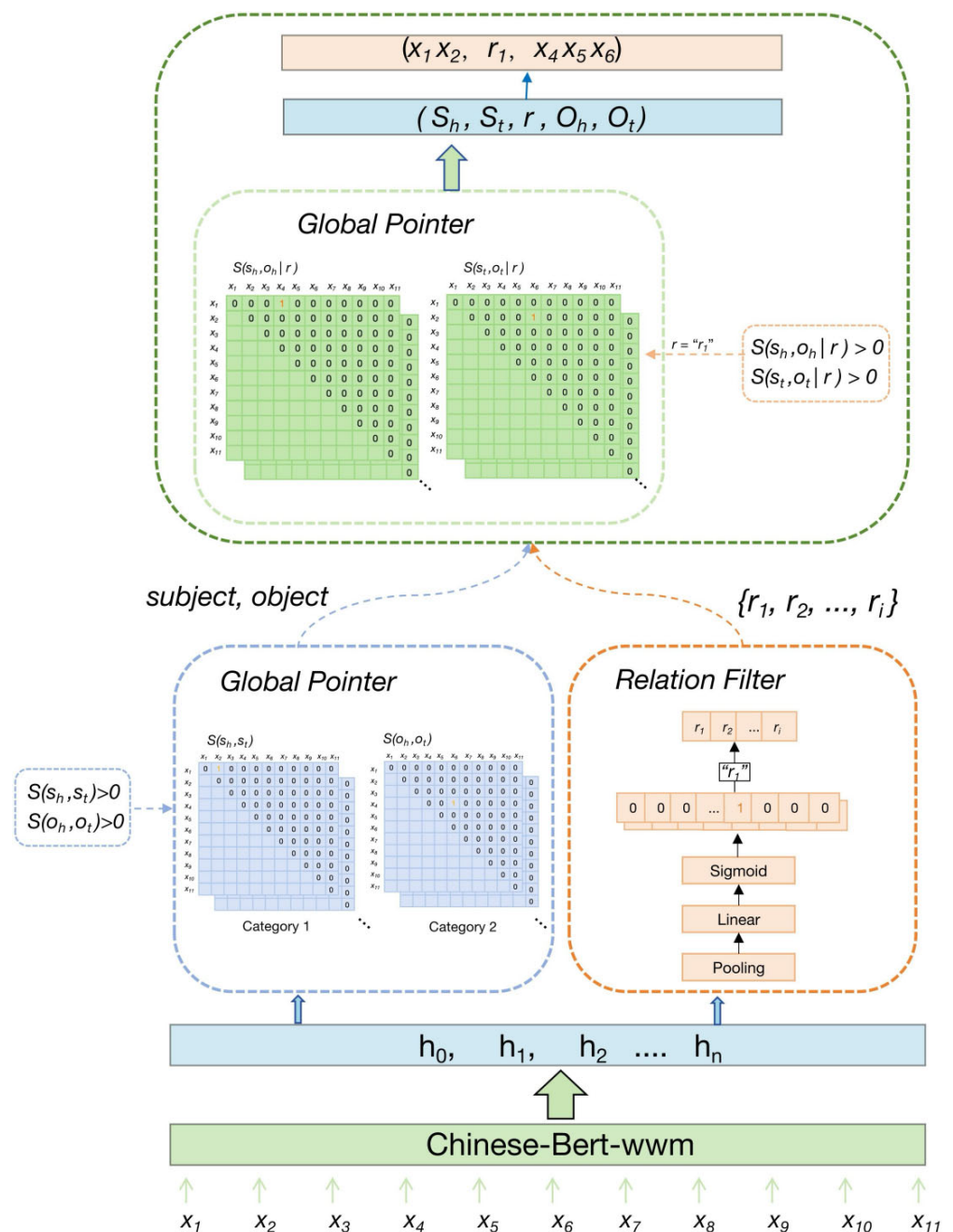


Figure 1. Structure of the GPRL.

### 3.2. GlobalPointer Decoding Module

This research utilizes a global pointer network to annotate entities, and the net can treat the start and end positions of entities as a whole, which can distinguish them and offer a more global perspective. The matrix labeling approach of the global pointer network addresses the issues of overlapping triplets and nested entities. A method called Rotary Position Embedding (RoPE) is used to design the scoring matrix using a dot-product attention mechanism and relative position encoding [38]. The RoPE enhances sensitivity to the length and span of entities, making it easier to differentiate between genuine entities. The matrix  $[h_1, h_2, \dots, h_n]$  obtained at the encoding layer undergoes the following transformations:

$$q_{i,a} = w_{q,\alpha}h_i + b_{q,\alpha}, \tag{2}$$

$$k_{i,a} = w_{k,\alpha}h_i + b_{k,\alpha} \tag{3}$$

where  $1 \leq i \leq n$ ,  $w_{q,\alpha}$  represents the weight parameter of the current linear transformation,  $b_{q,\alpha}$  is the bias term,  $w_{k,\alpha}$  represents the weight parameter of the current linear transformation, and  $b_{k,\alpha}$  is the bias term. The resulting sequences  $[q_{1,\alpha}, q_{2,\alpha}, \dots, q_{n,\alpha}]$  and  $[k_{1,\alpha}, k_{2,\alpha}, \dots, k_{n,\alpha}]$  are vector sequences used for identifying entities of type  $\alpha$ . The scoring function is defined at this point as:

$$s_\alpha = q_{i,\alpha}^T k_{j,\alpha} \tag{4}$$

$s_\alpha$  indicates the scoring of a continuous segment from  $i$  to  $j$  as an entity of type  $\alpha$ , using the dot product of  $q_{i,\alpha}$  and  $k_{j,\alpha}$  as the score for the entity segment. Subsequently, incorporating relative position information involves introducing a transformation matrix  $R_i$  that satisfies the relation  $R_i^T R_j = R_{j-i}$  and applying it to  $q$  and  $k$  to obtain the scoring function infused with relative positions  $s_a(i, j)$ :

$$s_a(i, j) = (R_i q_{i,\alpha})^T (R_j k_{j,\alpha}) = q_{i,\alpha}^T R_i^T R_j k_{j,\alpha} = q_{i,\alpha}^T R_{j-i} k_{j,\alpha} \tag{5}$$

where  $R_i, R_j$  are the transformation matrices used to add the relative position information for the satisfaction relation  $R_i^T R_j = R_{j-i}$ .

The task of entity relation extraction entails extracting triples in the form of <subject, relation, object>, i.e., (s, r, o). However, in practice, it involves extracting quintuples <subject head, subject tail, relation, object head, object tail>, i.e., (sh, st, r, oh, ot). From a probabilistic graph perspective, it suffices to design a scoring function  $S(\text{sh}, \text{st}, \text{r}, \text{oh}, \text{ot})$  for the quintuples. During training, annotated quintuples are set to be greater than zero, while the remaining ones are less than zero. All quintuples are then enumerated, and the part where the quintuple is greater than zero is outputted. Nevertheless, directly enumerating all quintuples leads to many possibilities. Assuming a sentence length of  $m$  and  $n$  relationships, the total number of quintuples is given by

$$n \times \frac{m(m+1)}{2} \times \frac{m(m+1)}{2} = \frac{1}{4}nm^2(m+1)^2 \tag{6}$$

Simplification of the above equation is achieved based on the following decomposition strategy:

$$S(\text{sh}, \text{st}, \text{r}, \text{oh}, \text{ot}) = S(\text{sh}, \text{st}) + S(\text{oh}, \text{ot}) + S(\text{sh}, \text{oh} | \text{r}) + S(\text{st}, \text{ot} | \text{r}) \tag{7}$$

where  $S(\text{sh}, \text{st})$  and  $S(\text{oh}, \text{ot})$  represent the head and tail scorings of the subject and object entities, respectively. If both parameters are greater than zero, it indicates the presence of both subject and object entities.  $S(\text{sh}, \text{oh} | \text{r})$  signifies relationship matching based on the head features of the subject and object entities. When there are no nested entities and  $S(\text{sh}, \text{oh} | \text{r}) > 0$ , all relationships can be determined. When there are nested entities, consideration is required for  $S(\text{st}, \text{ot} | \text{r})$ , the tail features of the subject and object entities.

### 3.3. Relation Filtering Decoding Module

The entities and relations extracted by the global pointer network are redundant relations, which shows that the number of relations identified in a sentence is much lower than the expected number. To address such issues, this research integrates a relation filtering decoding module into the GPLinker model. This module allows for the filtering and selection of predefined relations, which can improve the accuracy of entity relations identified by the global pointer network. The specific operational process is illustrated in the structural diagram.

Given a predefined set of relations  $R = \{r_1, r_2, \dots, r_k\}$  (with  $k$  denoting the size of set  $R$ ), the embedding vectors  $h_i$  corresponding to each character obtained from the output  $V = (h_1, h_2, \dots, h_n)$  at the encoding layer are subjected to pooling:

$$h_i^f = \text{Pooling}(h_i) \quad (8)$$

where  $h_i$  represents the corresponding embedding vector of each character.  $h_i^f$  represents the result after pooling of vector  $h_i$ ,  $h_i^f \in \mathbb{R}^{d \times k}$ ,  $d$  represents the embedding dimension.

Subsequently, the pooled results are passed through a linear layer and a non-linear activation function to obtain relation probabilities:

$$P^r = \text{Sigmoid}(W^r h_i^f + b^r) \quad (9)$$

where  $P^r \in \mathbb{R}^k$  denotes the probability of the text regarding relations, Sigmoid represents the non-linear activation function with a range of  $(0, 1)$ ,  $W^r \in \mathbb{R}^{d \times k}$  denotes the trainable weights, and  $b^r \in \mathbb{R}^k$  represents the bias term.

Finally, a threshold  $\mu$  is set to filter out redundant relations, and it can be taken as a multi-label binary classification task. If the relation probability  $P^r$  is greater than  $\mu$ , it is considered a relation to be retained; if it is less than  $\mu$ , it is deemed a redundant relation.

### 3.4. Loss Function

For entity extraction, the task involves extracting the heads and tails of subjects and objects for a specific relation. This model employs a sparse version of the multi-label cross-entropy loss function, which compares the scores of target categories with non-target categories to balance the weight of each category and address the issue of class imbalance [39]. The sparse version of the function reduces the transmission cost by only transmitting the indices corresponding to positive classes, instead of transmitting all the negative classes. The formula for this method is represented as follows:

$$\begin{aligned} L &= \log(1 + \sum_{i \in P} e^{-s_i}) + \log(1 + \sum_{i \in N} e^{s_i}) \\ &= \log(1 + \sum_{i \in P} e^{-s_i}) + \log(1 + \sum_{i \in A} e^{s_i}) + \log(1 - e^{\log(\sum_{i \in P} e^{s_i}) - \log(1 + \sum_{i \in A} e^{s_i})}) \end{aligned} \quad (10)$$

where  $L$  represents the loss of the global pointer network decoding module,  $P$  and  $N$  are the sets of positive and negative classes,  $A = P \cup N$ , and  $P$  calculates the loss corresponding to negative classes, while the loss for positive classes remains unchanged.  $s_i$  denotes the predicted value for the  $i$ th position in the label. The entity extraction task is divided into entity extraction, extraction of subject heads for a specific relation, and extraction of object tails for a specific relation. Since these three tasks are similar, they all use Formula (10) for the loss function. Therefore, the total loss combines the loss functions of the three tasks directly, as follows:

$$L_{GP} = L_{GP\_entity} + L_{GP\_head} + L_{GP\_tail} \quad (11)$$

where  $L_{GP}$  represents the total loss,  $L_{GP\_entity}$  represents the loss value for entity extraction,  $L_{GP\_head}$  represents the loss value for the head entity based on a specific relation, and  $L_{GP\_tail}$  represents the loss value for the tail entity based on a specific relation.

For the relation filtering task, a cross-entropy loss function is employed:

$$L_{\text{rel}} = -\frac{1}{k} \sum_{i=1}^k [y_i \log(P_i^r) + (1 - y_i) \log(1 - P_i^r)] \quad (12)$$

where  $L_{\text{rel}}$  denotes the loss of the relation filtering decoding module and  $K$  represents the total number of relations in the predefined relation set.  $y_i$  represents the true label of the relation  $r$ , and  $P_i^r$  represents the probability corresponding to the  $i$ th relation.

Since entity extraction and relation filtering are distinct tasks, this research adopts a loss function optimization strategy using the uncertainty weighting method from reference [32] to adaptively adjust the weights of the loss functions for these two tasks. This strategy allows the weights of the loss functions for the two tasks to be adaptively adjusted, based on maximizing the likelihood of Gaussian uncertainty with the same variance. Tasks with low noise and high certainty are assigned higher weights, enabling adaptive weight adjustments during model training and thereby saving time on parameter tuning. The derived multi-task loss function is as follows:

$$L(W, \Sigma_1, \Sigma_2) = \frac{1}{\Sigma_1^2} L_{\text{GP}}(W) + \frac{1}{\Sigma_2^2} L_{\text{REL}}(W) + \text{LOG} \Sigma_1 \Sigma_2 \quad (13)$$

where  $\sigma_1$  and  $\sigma_2$  represent the noise parameters for the entity relation extraction by the global pointer and for the relation filtering.

## 4. Experiments

### 4.1. Data and Evaluation Metrics

In this research, in order to verify the effectiveness of the model proposed for Chinese entity relation extraction, experiments were conducted on the Chinese medical information extraction dataset CMeIE and the Baidu dataset DuIE2.0. The CMeIE dataset is a Chinese medical information extraction dataset released in CHIP2020 [40]. The dataset mainly focuses on pediatric diseases and one hundred common diseases, and it contains nearly 75,000 triples. The total number of training and validation sets is 17,924, which are randomly divided into training, validation, and test sets. In addition, the DuIE2.0 dataset was used to further validate the effectiveness of the model proposed in this research on other domain Chinese datasets. The dataset is from the 2020 Language and Intelligence Technology Competition, primarily sourced from Baidu Baike and Baidu News, which contains nearly 430,000 triples, 210,000 Chinese sentences, and 48 predefined relation types [41]. In order to see the allocation and share of different data more clearly, the specific statistical analysis of the two datasets is shown in Table 1. The approximate distribution is in the ratio of 6:2:2.

**Table 1.** Statistical analysis of the dataset.

Dataset	Train (Ratio%)	Test (Ratio%)	Validation (Ratio%)	All	Relation
DuIE2.0	137,258 (68%)	34,044 (17%)	30,485 (15%)	201,787	48
CMeIE	11,651 (65%)	3584 (20%)	2689 (15%)	17,924	53

Both datasets exhibit complex overlapping and nesting issues. Therefore, using these two datasets can effectively verify the model proposed in this research in addressing the issues of relation overlap and entity nesting. The overlapping triple types mainly include normal, SEO, and EPO, as well as cases of nested entities (subject object overlap, SOO). The number and ratios of the four complex data types for the two datasets are shown in Table 2. From the data, it can be seen that the normal and SEO types have the highest percentage of data, occupying a major part of the dataset. In contrast, the EPO and SOO types of data have a lower percentage, which undoubtedly increases the difficulty of training. Nonetheless, the performance of the model is inferior compared with normal and SEO types in identifying EPO- and SOO-type data. However, the model can still accurately identify EPO- and SOO-type data with a relatively small amount of data, which fully demonstrates the practicality and effectiveness of our model.



**Table 2.** Quantitative statistics of the four complex data types.

Category	CMeIE			DuIE2.0		
	Train (Ratio%)	Test (Ratio%)	Validation (Ratio%)	Train (Ratio%)	Test (Ratio%)	Validation (Ratio%)
Normal	4375 (38%)	1329 (37%)	993 (37%)	87,591 (64%)	33,080 (97%)	27,043 (89%)
SEO	9678 (83%)	2979 (83%)	2257 (84%)	56,912 (41%)	704 (2%)	3168 (10%)
EPO	51 (0.4%)	12 (0.3%)	14 (0.5%)	6891 (5%)	126 (0.3%)	478 (1.5%)
SOO	1163 (9%)	375 (10%)	281 (10%)	2064 (2%)	273 (0.8%)	301 (0.9%)

Examples of the four complex data types are shown in Table 3. Generally, normal data are the simplest, followed by SEO data, SOO data, and EPO data. Therefore, EPO data have a smaller volume and are more difficult to extract.

**Table 3.** Examples of four complex data types.

Type	Example	Ternary
Normal	“Hepatic hydrothorax may cause breathing difficulties”	(Hepatic hydrothorax, clinical manifestations, dyspnea)
SEO	“Septic arthritis often involves the hips, knees, and ankles, and symptoms can last for several days”.	(Septic arthritis, hip) (Septic arthritis, knee) (Septic arthritis, ankle)
EPO	“Obstructive shock is due to decreased cardiac output not myocardial under function. Causes include pneumothorax, cardiac tamponade, pulmonary embolism, and aortic constriction, which results in obstruction of extracardiac blood flow pathways”.	(Obstructive shock, pathophysiology, pulmonary embolism) (Obstructive Shock, Related (Causes), Pulmonary embolism)
SOO	“Gallbladder inflammation abdominal mass may be palpable in the distended gallbladder in 30 to 40 percent of cases”.	(Gallbladder inflammation, site of onset, gallbladder)

In this research, the performance of the constructed entity–relationship joint extraction model is evaluated with entity–relationship extraction evaluation metrics, including precision (P), recall (R), and F1 value. The formulas are as follows:

$$P = \frac{TP}{TP + FP} \quad (14)$$

$$R = \frac{TP}{TP + FN} \quad (15)$$

$$F1 = \frac{2PR}{P + R} \quad (16)$$

where TP denotes the number of correctly predicted triples, FP denotes the number of incorrectly predicted triples, and FN denotes the number of true triples not correctly predicted. The joint entity–relationship extraction model constructed in this research identifies a correctly predicted triad only if the head entity, relationship, and tail entity are all correctly predicted.

#### 4.2. Setup

In the experiments, Python 3.8 and the PyTorch framework in deep learning were used. Model training and testing were mainly conducted on a Windows operating system with 64 GB memory, a 13th Gen Intel(R) Core(TM) i9-13900KF CPU, and an NVIDIA GeForce RTX 4090 GPU. According to the hardware conditions in the laboratory, the Chinese-Bert-WWM model was selected to obtain sequence feature vectors containing contextual information. The Adam optimizer was employed, and the specific parameter values in our model are shown in Table 4.

**Table 4.** Statistics of our model parameter values.

Parameter	Value	Meaning
maxlen	300	Truncation length of the text
batch_size	16	Batch size of training data
epochs	100	Number of rounds of training iterations
learning rate	$2 \times 10^{-5}$	Initial learning rate
pool_kernel_size	2	Pooling layer size
gp_threshold	0	Entity–relationship extraction threshold
relation_threshold	0.9	Relationship filtering threshold
relation_adding	0.3	Relationship filtering threshold

#### 4.3. Comparative Models

To evaluate the performance of the model proposed, comparisons were made with the following state-of-the-art entity relation extraction models. Choosing these models for comparison allows for a comprehensive and in-depth evaluation of the performance of the model. It also highlights the innovative points and advantages of our model through the comparison with different technical approaches and methods:

- **NovelTagging:** This model transforms the entity relation extraction problem into a sequence labeling problem, enabling the simultaneous recognition of entities and relations. While it does not address the issue of overlapping triples, it provides a foundational reference for our research.
- **CasRel:** Based on sequence labeling, this model introduces a cascaded binary tagging framework to tackle the problem of overlapping triples, which has become a critical milestone in the field.
- **TPLinker\_plus:** This model proposed a novel tagging scheme, which not only resolves overlapping triple issues, but also addresses entity nesting problems.
- **PRGC:** Based on a table-filling approach, this model introduces a new end-to-end framework that can resolve overlapping triple problems by separately predicting entities and relations. The model reduces redundant information generation and offers us a different perspective.
- **GPLinker:** Utilizing a global pointer-based model with matrix labeling, the model enables entity relation prediction from a global perspective. It successfully tackles overlapping triple issues and also resolves entity nesting problems, and it is taken as an important reference model for us.

In the context of the aforementioned model experiments, the specific parameters are detailed in Table 5. Given the substantial memory consumption of the TPLinker-plus and PRGC models, when the batch size was set to 16, the models encountered a memory overflow error, exceeding the available memory capacity of the current runtime environment. So, the batch size was adjusted to 8 for both TPLinker-plus and PRGC models. All other parameters were kept consistent with those described in our model, thereby facilitating a more equitable execution of the comparative experiments.

**Table 5.** Statistics of comparison models' parameter values.

Parameter	NovelTagging	CasRel	TPLinker-Plus	PRGC	GPLinker
maxlen	300	300	300	300	300
batch_size	16	16	8	8	16
epochs	100	100	100	100	100
learning rate	$2 \times 10^{-5}$	$2 \times 10^{-5}$	$2 \times 10^{-5}$	$2 \times 10^{-5}$	$2 \times 10^{-5}$

## 5. Results and Discussion

### 5.1. Model Comparative Experiment

The comparative experimental results of the model proposed and several existing baseline models are for the CMeIE dataset and DuIE2.0 dataset (Table 6.)

**Table 6.** Comparison of model experiment results.

Model	CMeIE			DuIE2.0		
	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
NovelTagging	51.4	17.1	25.6	60.6	45.3	55.1
PRGC	40.8	46.4	43.4	48.8	81.4	61.3
CasRel	50.5	48.5	49.5	62.0	80.4	70.6
TPLinker_plus	50.7	49.8	50.2	64.3	74.4	69.0
GPLinker	51.2	47.5	49.6	61.7	81.0	70.4
GPRL (our model)	54.7	50.6	51.7	64.1	81.6	71.8

Table 6 shows that the GPRL model proved effective on the CMeIE dataset, achieving a precision of 54.7%, a recall of 50.6%, and an F1 score of 51.7%. A comparative analysis shows that the entity relation joint extraction model outperforms traditional pipeline methods like NovelTagging, which demonstrates the issue of error propagation in pipeline approaches. The CasRel model transforms relations into a function mapping from head entities to tail entities, accurately identifying more entities in sentences. However, it falls short of recognizing nested entities, resulting in lower accuracy compared to the model in this work. TPLinker\_plus can solve the nested entity problem but suffers from a more complex model structure leading to resource wastage, ultimately yielding lower recall and F1 scores. PRGC combines the strengths of CasRel and TPLinker. Inspired by PRGC's relation filtering, the GPRL model applies a similar approach. PRGC still inherits the complexity issues of TPLinker, leading to resource wastage, longer training times, and lower effectiveness. GPRL builds upon GPLinker by incorporating relation filtering and adopting a multi-task learning strategy to balance task losses. It achieves better precision, recall, and F1 score, with improvements of 26.1%, 8.3%, 2.2%, and 1.5% over other baseline models and a 2.1% improvement over the original GPLinker model in F1 score.

To further validate the effectiveness of the model on Chinese datasets, comparative experiments were conducted on the DuIE2.0 dataset compared with other baseline models (Table 6). The GPRL model proves effective on the DuIE2.0 dataset, achieving a precision of 64.1%, a recall of 81.6%, and an F1 score of 71.8%. Compared to other baseline models, there are improvements of 16.7%, 10.5%, 2.8%, and 1.2% in F1 score, as well as a 1.4% improvement over the original GPLinker model. The original GPLinker model already excelled in model structure, training strategies, training speed, and results compared with other baseline models, thereby demonstrating the significance of the improvements made in this work to the GPLinker model.

Furthermore, the experimental analysis indicates that model performance on the CMeIE dataset is generally lower due to the dataset's domain specificity in the medical field. The Chinese-Bert-WWM pre-trained language model used was trained on large-scale general corpora, leading to difficulties in semantic understanding of specialized medical data. The examples of unpredicted correct triples are shown in Table 7. In the triple (submucous cleft palate, related symptom, airway obstruction), "submucous cleft palate" is a rare medical term that the entity recognition failed to identify, resulting in the triple being unable to be predicted correctly. In the triple (gastritis (Chinese), Synonym, gastritis (English)), the Chinese expression for gastritis was not recognized as a synonym for its English expression. This is because Chinese-Bert-WWM is trained on a Chinese corpus, and its recognition of English terms needs to be improved. Hence, the Chinese-Bert-WWM model may exhibit a poor representation of medical data, resulting in errors or missed extractions of many triples.

**Table 7.** Examples of unpredicted correct triples.

Examples of Unpredicted Correct Triples	Example
(Diseases, Related (Symptoms), Diseases) (Symptoms, Synonyms, Symptoms)	(submucous cleft palate, related (symptom), airway obstruction) (gastritis (Chinese), Synonym, gastritis (English))

To demonstrate the contribution of GPRL, it was compared with several existing studies on entity and relation extraction based on the datasets CMeIE and DuIE2.0 (Table 8). Despite variations in experimental environment configurations and parameters, the performance of GPRL is better than most models, as indicated by the P, R, and F1 values, which also suggests that GPRL outperforms some of the existing research results.

**Table 8.** Comparison of existing research.

Model	CMeIE			DuIE2.0		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
HNNERJE [42]	48.49	46.37	47.40	/	/	/
FA-CBT [43]	54.51	48.63	51.40	/	/	/
DEPR [44]	47.51	46.11	46.80	71.06	65.35	68.09
REACT [45]	/	/	/	68.5	66.0	67.2
CasRelBLCF [46]	/	/	/	74.0	68.6	71.2
GPRL (our model)	54.7	50.6	51.7	64.1	81.6	71.8

## 5.2. Ablation Experiment

To analyze the contributions of each module in the entity relation joint extraction model constructed in this research, relevant ablation experiments were conducted on CMeIE and DuIE2.0 datasets. The performance indicators of different modules are shown in Table 9. The results indicate that both the relationship filtering and the loss function optimization integrated into this model have been proven to be effective.

**Table 9.** Results of ablation experiments.

Model	CMeIE			DuIE2.0		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
GPlinker	51.2	47.5	49.6	61.7	81.0	70.4
GPlinker + Relation filter	53.5	48.3	50.8	63.6	81.3	71.5
GPlinker + Relation filter + Loss optimization	54.7	50.6	51.7	64.1	81.6	71.8

Relationship filtering can reduce relationship redundancy and improve the model's prediction accuracy. Table 9 shows that the relationship filtering component increased precision by 2.3%, recall by 0.8%, and F1 score by 1.2% on the CMeIE dataset. On DuIE2.0 dataset, precision increased by 1.9%, recall by 0.3%, and F1 score by 1.1%. The main reason is that predicting triplets requires judgment on many relationships, and the original model consumes more resources and time, leading to a decrease in accuracy. Optimizing the loss function dynamically adjusts the weights of multiple tasks based on noise during the model training process, ultimately enhancing model performance. As shown in Table 9, adding the loss function optimization component increased precision by 1.2%, recall by 2.3%, and F1 score by 0.9% on the CMeIE dataset. On the DuIE2.0 dataset, precision increased by 0.5%, recall by 0.3%, and F1 score by 0.3%. The results show that adding components to GPRL on different datasets led to improvements, with the relationship filtering module having a greater impact on GPRL and the effect of loss function optimization being relatively smaller but still contributing to enhancements, indicating that both modules are effective for GPRL.

### 5.3. Overlapping Triples and Nested Entities Experiment

To validate the effectiveness of the model in addressing overlapping triples, further experiments were conducted on CMeIE and DuIE2.0 datasets. The dataset’s test data were partitioned into four subsets based on different overlapping types: normal, EPO, SEP, and SOO. Experiments were conducted on each of these four sub-datasets. The experimental results are depicted in Figure 2.

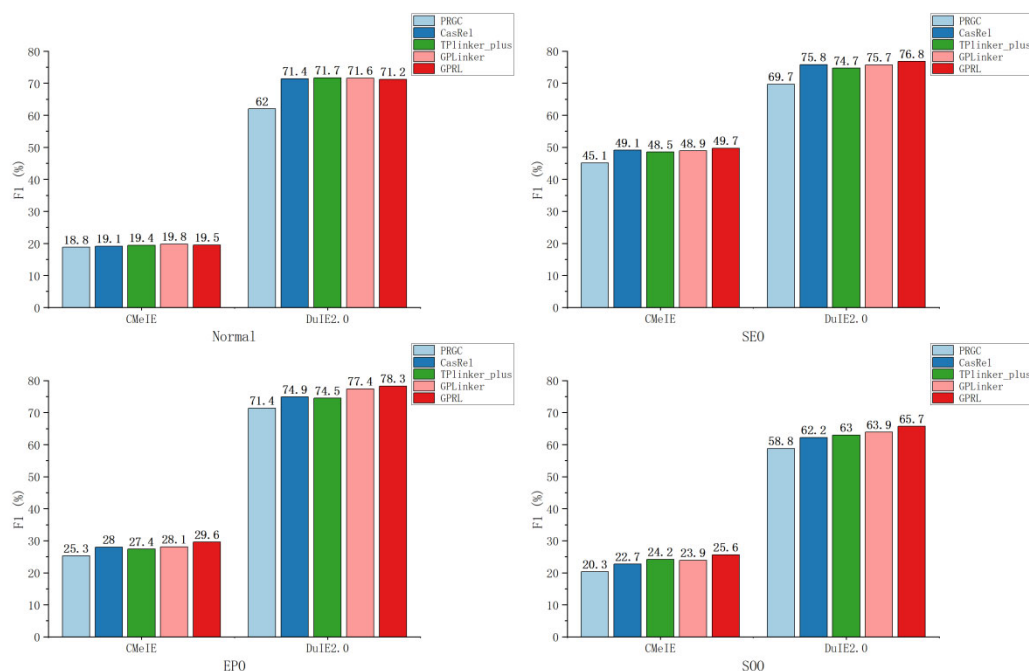


Figure 2. Overlapping ternary and nested entity result diagrams.

The analysis of the CMeIE dataset reveals that the SEO data show normal levels, while the EPO, SOO, and normal data exhibit an overall lower performance. Such a phenomenon may stem from the uneven distribution of the dataset, with a higher prevalence of SEO data and a scarcity of other datasets, leading to lower overall results. However, The GPRL model’s performance is compromised when dealing with normal-type data, possibly due to error propagation. In this scenario, the relationship filter might extract inaccurate relationships, leading to incorrect ternary extractions. Despite this setback, the F1 values achieved for other types of overlapping data surpass those of other models. This outcome highlights the effectiveness of GPRL in addressing overlapping and nested entity challenges.

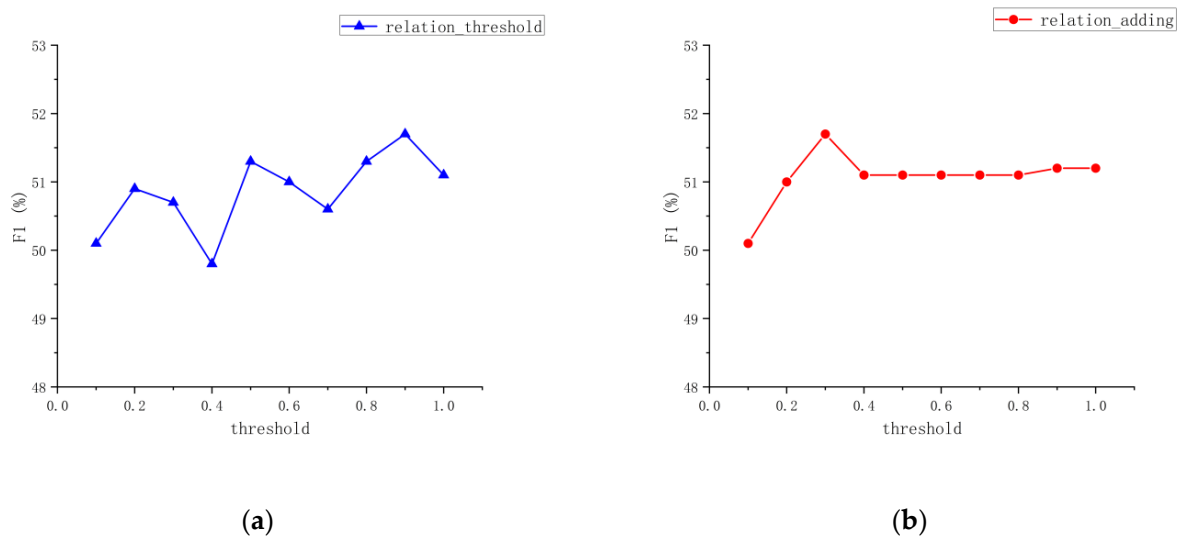
In the analysis of the DuIE2.0 dataset, it is observed that all data types are within normal ranges. This points out that the lower performance in the CMeIE dataset is due to the dataset itself rather than the model. When normal triples are extracted from DuIE2.0, performance drops again. When SEO triples are extracted, the model’s performance is slightly better than the baseline model, and a significant improvement is observed over the baseline model when EPO triples are extracted. EPO involves entity pair overlap, which is a more complex form of overlap compared to SEO. This suggests that other baseline models exhibit decreased performance when handling more complex overlap scenarios, while GPRL performs better in such intricate situations. Furthermore, GPRL also stands out in extracting SOO triples, proving its ability to recognize nested entities better than others. Nevertheless, all models displayed a decreasing trend in addressing SOO challenges, indicating that there is still ample room for improvement in the recognition of nested entities.

In the training process, GPRL and GPLinker presented better in the training process, while TPLinker\_plus and PRGC incurred high training costs due to their intricate model structures. This further underscores the superiority of GPRL over other baseline models

across various aspects. The GPRL model proposed in this research excels in addressing overlapping ternary and nested entity challenges, demonstrating efficient and accurate extraction of ternary information compared to its counterparts.

#### 5.4. Parameter Experiment

In this research, further investigation was conducted to explore the impact of different relation filtering thresholds on model performance and determine the optimal model parameters. The results of experiments performed on the CMeIE dataset by adjusting relation filtering thresholds are presented in Figure 3, where 'relation\_threshold' represents the threshold for determining relation filtering and 'relation\_adding' represents the threshold for adding after relation filtering to the original model.



**Figure 3.** Parameter change curve. (a) The threshold for determining relation filtering; (b) the threshold for adding after relation filtering to the original model.

Figure 3a shows that when 'relation\_adding' remains fixed, the 'relation\_threshold' fluctuates up and down, reaching a peak at 0.9 before decreasing. Figure 3b shows that when 'relation\_threshold' is fixed, 'relation\_adding' reaches a peak at 0.3 before stabilizing around 51.7%. It is evident that these two thresholds do not exhibit a consistent or sensitive pattern with the overall model. Therefore, after conducting threshold parameter comparison experiments, this work selected the optimal parameter values of 'relation\_threshold' as 0.9 and 'relation\_adding' as 0.3 to achieve the best performance for the model.

#### 5.5. Discussion

Based on the experimental results above, the following advantages of GPRL can be stated: Compared with traditional entity relationship extraction models, it does not require a significant amount of manual involvement in feature extraction, thus saving valuable time and resources. In the comparative experiments, GPRL outperformed the traditional model NovelTagging. Additionally, compared with the common pointer annotation entity relationship extraction model, GPRL can identify overlapping triples and nested entities with a global view and performs entity relationship extraction accurately. In the nested data experiment, GPRL outperformed the pointer-based model CasRel in terms of results. The model is specifically designed for Chinese text, maximizing contextual learning and inference within Chinese language data, and presents advantages such as superior efficiency, cost-effectiveness, and better results compared with other baseline models.

However, GPRL faces challenges and limitations that require further improvement. It may have difficulty in extracting specialized domain knowledge, particularly in domains with complex and unique entity relations, which require more meticulous handling and

learning. The main reasons are as follows: GPRL's performance may suffer when dealing with niche or specialized domains due to a lack of sufficient training data in those areas. Complex entity relationships that are ambiguous or context-dependent can pose challenges for GPRL in accurately capturing and understanding the underlying connections. Additionally, the model may encounter limitations in handling long-distance dependency relationships in lengthy texts or complex contexts, necessitating enhanced mechanisms for capturing global contextual information. The model's performance may also be impacted by data scarcity and annotation quality, requiring enhanced robustness and generalization for low-frequency relations or noisy data.

## 6. Conclusions

The proposed model GPRL, based on relation filtering and multi-task learning for entity relation extraction, was introduced in this research. The model utilizes a global pointer network to transform the extraction of triples into quintuples and converts the extraction of entity heads and tails under specific relations into a form similar to scaled dot-product attention. GPRL can identify overlapping triples and nested entities by employing multiple specific relation matrices. Traditional pointer networks typically suffer from the inability to consider entity positions globally, leading to inconsistencies between training and prediction. On the other hand, the global pointer network used in GPRL enhances model performance by enabling holistic discrimination of entity positions, which ensures consistency between training and prediction objectives. In addition, the implementation of relation filtering within the global pointer network considerably reduces the necessity for repetitive relation judgments. This approach enhances the model's accuracy and efficiency by eliminating the requirement for manually annotating entities for each predefined relation in the dataset, leading to more accurate triple extraction. Further, the use of a multi-task learning strategy successfully handles inter-task imbalances, which is particularly beneficial in situations with limited training data. This strategy also mitigates the challenge of overfitting, resulting in significant improvement in the model's performance. In comparison experiments conducted on the publicly available Chinese datasets CMeIE and DuIE2.0, as well as complex and ablation experiments, GPRL consistently outperformed current baseline models and proved state-of-the-art performance in Chinese datasets for entity relation extraction.

Future research will focus on advancing entity relation extraction by developing innovative approaches, refining model architectures, and optimizing training strategies to improve generalizability and performance. We aim to enhance practical applications and build knowledge graphs and intelligent Q&A systems that can understand user queries, retrieve accurate answers, and engage in complex reasoning. This will elevate user experience and expand AI applications in sectors like customer service, education, healthcare, and environmental resources.

**Author Contributions:** Conceptualization, B.L.; methodology, W.C.; software, W.C.; validation, Y.Z. and W.C.; formal analysis, J.T.; investigation, L.H.; resources, L.H.; data curation, Y.Z. and M.C.; writing—original draft preparation, J.T.; writing—review and editing, B.L.; visualization, J.T.; supervision, L.H.; project administration, D.T.; funding acquisition, D.T. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by The Major Science and Technology Projects of Sichuan Province, grant number 2022ZDZX0001, The Science and Technology Support Project of Sichuan Province, grant number 2023YFS0366 and 2023YFG0020, and Natural Science Foundation of Sichuan Province, grant number 2024NSFSC0792.

**Data Availability Statement:** The CMeIE dataset that supports the findings of this study is openly available in [TIANCHI Data Platform] at <https://tianchi.aliyun.com/dataset/95414> (22 March 2021). The DuIE2.0 dataset that supports the findings of this study is openly available in [AI Studio Data Platform] at <https://aistudio.baidu.com/datasetdetail/180082> (28 November 2022).

**Conflicts of Interest:** The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

## References

1. Ji, S.; Pan, S.; Cambria, E.; Marttinen, P.; Philip, S.Y. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 494–514. [[CrossRef](#)] [[PubMed](#)]
2. Tan, L.; Haihong, E.; Kuang, Z.; Song, M.; Liu, Y.; Chen, Z.; Xie, X.; Li, J.; Fan, J.; Wang, Q.; et al. Key technologies and research progress of medical knowledge graph construction. *Big Data Res.* **2021**, *7*, 80–104.
3. Cheng, D.; Yang, F.; Wang, X.; Zhang, Y.; Zhang, L. Knowledge graph-based event embedding framework for financial quantitative investments. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual, 25–30 July 2020; pp. 2221–2230.
4. Wang, Q.; Mao, Z.; Wang, B.; Guo, L. Knowledge graph embedding: A survey of approaches and applications. *IEEE Trans. Knowl. Data Eng.* **2017**, *29*, 2724–2743. [[CrossRef](#)]
5. Liu, H.; Jiang, Q.; Gui, Q.; Zhang, Q.; Wang, Z.; Wang, L.; Wang, L. A Review of Research Progress in Entity Relationship Extraction Techniques. *Comput. Appl. Res.* **2020**, *37*, 1–5.
6. Chowdhary, K.; Chowdhary, K.R. Natural language processing. In *Fundamentals of Artificial Intelligence*; Springer: New Delhi, India, 2020; pp. 603–649.
7. Li, D.; Zhang, Y.; Li, D.; Lin, D. A survey of entity relation extraction methods. *J. Comput. Res. Dev.* **2020**, *57*, 1424–1448.
8. Feng, J.; Zhang, T.; Hang, T. An Overview of Overlapping Entity Relationship Extraction. *J. Comput. Eng. Appl.* **2022**, *58*, 1–11.
9. Hong, E.; Zhang, W.; Xiao, S.; Chen, R.; Hu, Y.; Zhou, X.; Niu, P. Survey of entity relationship extraction based on deep learning. *J. Softw.* **2019**, *30*, 1793–1818.
10. Zeng, D.; Liu, K.; Lai, S.; Zhou, G.; Zhao, J. Relation classification via convolutional deep neural network. In Proceedings of the COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, Dublin, Ireland, 23–29 August 2014; pp. 2335–2344.
11. Socher, R.; Huval, B.; Manning, C.D.; Ng, A.Y. Semantic compositionality through recursive matrix-vector spaces. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Jeju Island, Republic of Korea, 12–14 July 2012; pp. 1201–1211.
12. Yu, Y.; Si, X.; Hu, C.; Zhang, J. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput.* **2019**, *31*, 1235–1270. [[CrossRef](#)] [[PubMed](#)]
13. 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, Berlin, Germany, 7–12 August 2016; Short papers. Volume 2, pp. 207–212.
14. 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, Lisbon, Portugal, 17–21 September 2015; pp. 1785–1794.
15. Zhang, S.; Zheng, D.; Hu, X.; Yang, M. Bidirectional long short-term memory networks for relation classification. In Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation, Shanghai, China, 30 October–1 November 2015; pp. 73–78.
16. Luo, L.; Yang, Z.; Yang, P.; Zhang, Y.; Wang, L.; Lin, H.; Wang, J. An attention-based BiLSTM-CRF approach to document-level chemical named entity recognition. *Bioinformatics* **2018**, *34*, 1381–1388. [[CrossRef](#)]
17. Miwa, M.; Bansal, M. End-to-end relation extraction using lstms on sequences and tree structures. *arXiv* **2016**, arXiv:1601.00770.
18. Zheng, S.; Wang, F.; Bao, H.; Hao, Y.; Zhou, P.; Xu, B. Joint extraction of entities and relations based on a novel tagging scheme. *arXiv* **2017**, arXiv:1706.05075.
19. Yu, B.; Zhang, Z.; Shu, X.; Wang, Y.; Liu, T.; Wang, B.; Li, S. Joint extraction of entities and relations based on a novel decomposition strategy. *arXiv* **2019**, arXiv:1909.04273.
20. Wei, Z.; Su, J.; Wang, Y.; Tian, Y.; Chang, Y. A novel cascade binary tagging framework for relational triple extraction. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 1–13.
21. Wang, L.; Xiong, C.; Deng, N. A research on overlapping relationship extraction based on multi-objective dependency. In Proceedings of the 15th International Conference on Computer Science & Education (ICCSE), Delft, The Netherlands, 18–22 August 2020; pp. 618–622.
22. Xiao, L.; Zang, Z.; Song, S. Research on relational Extraction Cascade labeling framework incorporating Self-attention. *Comput. Eng. Appl.* **2023**, *59*, 77.
23. Wang, Y.; Yu, B.; Zhang, Y.; Liu, T.; Zhu, H.; Sun, L. Tplinker: Single-stage joint extraction of entities and relations through token pair linking. In Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain, 8–13 December 2020; pp. 1572–1582.
24. Zheng, H.; Wen, R.; Chen, X.; Yang, Y.; Zhang, Y.; Zhang, Z.; Zhang, N.; Qin, B.; Xu, M.; Zheng, Y. PRGC: Potential relation and global correspondence based joint relational triple extraction. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics, Bangkok, Thailand, 1–6 August 2021; pp. 6225–6235.



25. Su, J.; Murtadha, A.; Pan, S.; Hou, J.; Sun, J.; Huang, W.; Wen, B.; Liu, Y. Global pointer: Novel efficient span-based approach for named entity recognition. *arXiv* **2022**, arXiv:2208.03054.
26. Su, J. Gplinker: Entity Relationship Joint Extraction Based on Globalpointer. Available online: <https://spaces.ac.cn/archives/8888> (accessed on 30 January 2022).
27. Zhang, Y.; Liu, S.; Liu, Y.; Ren, L.; Xin, Y. A review of deep learning-based joint extraction of entity relationships. *Electron. Lett.* **2023**, *51*, 1093.
28. Michael, C. Multi-task learning with deep neural networks: A survey. *arXiv* **2020**, arXiv:2009.09796.
29. Zhao, F.; Jiang, Z.; Kang, Y.; Sun, C.; Liu, X. Adjacency List Oriented Relational Fact Extraction via Adaptive Multi-task Learning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021.
30. Chen, Z.; Badrinarayanan, V.; Lee, C.Y.; Rabinovich, A. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *Proceedings of the International Conference on Machine Learning (PMLR)*, Stockholm, Sweden, 10–15 July 2018; pp. 794–803.
31. Liu, S.; Johns, E.; Davison, A.J. End-to-end multi-task learning with attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, 15–20 June 2019; pp. 1871–1880.
32. Guo, M.; Haque, A.; Huang, D.A.; Yeung, S.; Fei-Fei, L. Dynamic task prioritization for multitask learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, Munich, Germany, 8–14 September 2018; pp. 270–287.
33. Kendall, A.; Gal, Y.; Cipolla, R. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7482–7491.
34. Cui, Y.; Che, W.; Liu, T.; Qin, B.; Yang, Z. Pre-training with whole word masking for Chinese bert. *IEEE ACM Trans. Audio Speech Lang. Process.* **2021**, *29*, 3504–3514. [[CrossRef](#)]
35. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2019**, arXiv:1810.04805.
36. Buckman, J.; Roy, A.; Raffel, C.; Goodfellow, I. Thermometer encoding: One hot way to resist adversarial examples. In *Proceedings of the International Conference on Learning Representations*, Vancouver, BC, Canada, 30 April–3 May 2018.
37. Church, K.W. Word2Vec. *J. Nat. Lang. Eng.* **2017**, *23*, 155–162. [[CrossRef](#)]
38. Su, J.; Ahmed, M.; Lu, Y.; Pan, S.; Bo, W.; Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing* **2024**, *568*, 127063. [[CrossRef](#)]
39. Su, J.; Zhu, M.; Murtadha, A.; Pan, S.; Wen, B.; Liu, Y. Zlpr: A novel loss for multi-label classification. *arXiv* **2022**, arXiv:2208.02955.
40. Guan, T.; Zan, H.; Zhou, X.; Xu, H.; Zhang, K. CMeIE: Construction and evaluation of Chinese medical information extraction dataset. In *Natural Language Processing and Chinese Computing, Proceedings of the 9th CCF International Conference, NLPCC 2020, Zhengzhou, China, 14–18 October 2020*; Proceedings, Part I 9; Springer International Publishing: Cham, Switzerland, 2020; pp. 270–282.
41. Li, S.; He, W.; Shi, Y.; Jiang, W.; Liang, H.; Jiang, Y.; Zhang, Y.; Lyu, Y.; Zhu, Y. Duie: A large-scale chinese dataset for information extraction. In *Natural Language Processing and Chinese Computing, Proceedings of the 8th CCF International Conference, NLPCC 2019, Dunhuang, China, 9–14 October 2019*; Proceedings, Part II 8; Springer International Publishing: Cham, Switzerland, 2019; pp. 791–800.
42. Mao, D.; Li, X.; Liu, J.; Zhang, D.; Yan, W. Chinese entity and relation extraction model based on parallel heterogeneous graph and sequential attention mechanism. *Comput. Appl.* **2024**, *44*, 2018–2025.
43. Lu, X.; Tong, J.; Xia, S. Entity relationship extraction from Chinese electronic medical records based on feature augmentation and cascade binary tagging framework. *Math. Biosci. Eng.* **2024**, *21*, 1342–1355. [[CrossRef](#)]
44. Xiao, Y.; Chen, G.; Du, C.; Li, L.; Yuan, Y.; Zou, J.; Liu, J. A Study on Double-Headed Entities and Relations Prediction Framework for Joint Triple Extraction. *Mathematics* **2023**, *11*, 4583. [[CrossRef](#)]
45. Kong, L.; Liu, S. REACT: Relation Extraction Method Based on Entity Attention Network and Cascade Binary Tagging Framework. *Appl. Sci.* **2024**, *14*, 2981. [[CrossRef](#)]
46. Tang, H.; Zhu, D.; Tang, W.; Wang, S.; Wang, Y.; Wang, L. Research on joint model relation extraction method based on entity mapping. *PLoS ONE* **2024**, *19*, e0298974. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.