



Article Entity Relationship Extraction Based on a Multi-Neural Network Cooperation Model

Yibo Liu *^(D), Qingyun Zuo, Xu Wang and Teng Zong

School of Information and Communication, National University of Defense Technology, Wuhan 430074, China; zuoqy@163.com (Q.Z.); wangxu9191@gmail.com (X.W.); zt@nudt.edu.cn (T.Z.) * Correspondence: liuyibo@nudt.edu.cn

Abstract: Entity relation extraction mainly extracts relations from text, which is one of the important tasks of natural language processing. At present, some special fields have insufficient data; for example, agriculture, the metallurgical industry, etc. There is a lack of an effective model for entity relationship recognition under the condition of insufficient data. Inspired by this, we constructed a suitable small balanced data set and proposed a multi-neural network collaborative model (RBF, Roberta–Bidirectional Gated Recurrent Unit–Fully Connected). In addition, we also optimized the proposed model. This model uses the Roberta model as the coding layer, which is used to extract the word-level features of the text. This model uses BiGRU (Bidirectional Gated Recurrent Unit)–FC (Fully Connected) as the decoding layer, which is used to obtain the optimal relationship of the text. To further improve the effect, the input layer is optimized by feature fusion, and the learning rate is optimized by the cosine annealing algorithm. The experimental results show that, using the small balanced data set, the F1 value of the RBF model proposed in the paper is 25.9% higher than the traditional Word2vec–BiGRU–FC model. It is 18.6% higher than the recent Bert–BiLSTM (Bidirectional Long Short Term Memory)–FC model. The experimental results show that our model is effective.

Keywords: natural language processing; entity relationship extraction; multi-neural network cooperation



Citation: Liu, Y.; Zuo, Q.; Wang, X.; Zong, T. Entity Relationship Extraction Based on a Multi-Neural Network Cooperation Model. *Appl. Sci.* 2023, *13*, 6812. https://doi.org/ 10.3390/app13116812

Received: 5 May 2023 Revised: 28 May 2023 Accepted: 31 May 2023 Published: 3 June 2023



Copyright: © 2023 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

In the current era of big data, all kinds of data and information are growing explosively [1], and artificial intelligence technology is developing rapidly with the support of sufficient data. However, the problem of how to make machines think like people has not been solved, there is still a gap between the development of perceptual intelligence and people's expectations. Knowledge graph technology can provide background knowledge for machine thinking, and become an important way to realize artificial cognitive intelligence. Entity relationship extraction is one of the key links in the construction of a knowledge graph.

Compared with English entity relationship extraction, Chinese entity relationship extraction has the characteristics of a relative lack of open research datasets and a more prominent importance of word-level features [2]. In addition, how to ensure a better effect of the model under the condition of a small dataset has always been a major challenge in the field of entity relationship extraction [3]. To solve the above problems, we propose a multi-neural network collaborative RBF model, which can achieve good results under the condition of small balanced samples. The multi-neural network cooperation model proposed in this paper is a new combination of multiple deep learning models. This paper also proposes an optimization method for the multi-neural network cooperation model.

The novel contributions of our work are summarized as follows:

(1) Proposing the idea of conducting a relationship extraction experiment under the condition of a balanced small sample. This paper constructs a small, balanced sample relationship extraction dataset and can judge which model is better.

(2) Aiming at the task of entity relationship extraction under the condition of a small balanced dataset, we studied several deep learning models and the effect of multi-model collaboration. Given this, this paper put forward a new multi-neural network cooperation model, namely the RBF (Roberta–BiGRU–FC) model, based on our research. The Roberta–BiGRU–FC model uses Roberta as the coding layer and BiGRU–FC as the decoding layer. The F1 value of the experimental result on the constructed small sample data set reached 89.6%.

(3) Proposing an effective optimization method for the RBF multi-neural network cooperation model. We optimized the RBF model on the input feature layer and learning rate to improve the effect on the task of extracting entity relations from small samples. The learning rate of the entity relationship extraction model is generally fixed, and it cannot be optimized adaptively according to the transformation of the loss function. The cosine annealing algorithm can make the learning rate dynamically adjusted, and the learning rate first slows down and then accelerates to decline. Given this, this paper applied the cosine annealing algorithm to the optimization of RBF models. After optimization, the F1 value on the constructed small sample dataset reached 91.9%.

The rest of this paper is organized as follows: the second section briefly introduces the relevant work, the third section introduces the overall architecture of the RBF model, the fourth section describes the process and results of the experiment, the fifth chapter analyzes the experimental results, and the sixth section summarizes the content of the paper and describes the next steps.

2. Related Work

How to extract the required triples (head-entity, relationship, tail-entity) from the current massive unstructured text and form a knowledge base has always been the focus of research in the field of knowledge graphs [4]. At present, the mainstream methods of triplet extraction include pipeline extraction and joint extraction. Pipeline extraction takes entity recognition and relationship extraction as different tasks and models in series. Joint extraction takes entity recognition and relationship extraction series. In contrast, joint extraction modeling is more complex and lacks flexibility compared with the pipeline method [5,6], so this paper mainly explores entity relationship recognition in pipeline extraction.

Methods for entity relation extraction can be summarized into three categories: template rules-based methods, traditional machine learning-based methods, and deep learningbased methods [7–9].

2.1. Entity Relationship Extraction Based on Template Rules

The method based on template rules was mainly intended to find the characteristics and laws of entity relations when they appear in the text and set a series of template rules by using regular expressions [10]. In 1998, Aone et al. proposed that experts design template rules [11], detect and match the text, and match the entity relations that conform to the template. Later, there were also methods such as using a syntactic analyzer to construct text dependencies [12] to improve the efficiency of building template rules. The method based on template rules often needs the experienced support of specific fields in specifying rules, and the rule templates in different fields often cannot be used mutually, which has characteristics of poor universality. It is difficult to use the same set of rules effectively across fields [13].

2.2. Entity Relation Extraction Based on Traditional Machine Learning

Most traditional machine learning models mainly focus on extracting features from sample data. These models usually use classical models such as SVM (Support Vector Machine), the Markov chain, and logistic regression [14]. In 2005, Zhou et al. [15] used the SVM machine learning model for entity relationship extraction, and the F1 value on the ACE database reached 55.5%. In 2006, Culotta, A. and others applied the Markov chain

model to the task of entity relationship extraction [16]. In 2009, Li [17] and others processed the data input layer of SVM and made some achievements in relationship extraction in the biomedical field by using word bag features, part-of-speech features, and dependency features as model inputs.

Due to the defects of traditional machine learning-based entity relationship extraction methods and the propagation of feature extraction errors, it is often difficult to obtain a good recall rate in small sample entity relationship extraction.

2.3. Entity Relationship Extraction Based on Deep Learning

With the rise of deep learning technology, more and more deep learning models, such as CNN (Convolutional Neural Network), RNN (Recurrent Neural Network), LSTM, and GRU, are used in entity relationship extraction tasks [18–21].

In 2013, Liu et al. first proposed using the CNN method for relationship classification and encoding the input words through the dictionary of synonyms. The model mainly included a convolution layer, full connection layer, and softmax classification [22], which inspired the subsequent deep learning model for relationship extraction. In 2014, Zeng et al. proposed using word vectors and position vectors as the input of a convolution neural network, which could better improve the effect of relationship extraction [19]. In 2015, Zhang et al. [23] proposed using RNN for relationship extraction, without using any lexical features; the effect was similar to that of CNN combined with lexical features. In 2016, Zhou et al. [24] proposed LSTM (long short-term memory) instead of the RNN model, added an attention mechanism, proposed BiLSTM attention model architecture, and used word vectors and position vectors as feature representations of the model input layer, which improved the effect of relationship recognition.

Most of the above studies used the traditional pre-training method [25] to vectorize the text. Among them, the Word2vec text feature extraction model [26] released by Google in 2013 had a good effect in the traditional method, but Word2vec is a static method and cannot be dynamically optimized for specific tasks.

In 2018, Google released Bert's pre-training model [27], which could obtain the dependence of dynamically encoded word vectors to capture longer distances by fully extracting words and sentences [28,29]. At the end of October 2018, it announced Bert's performance in 11 NLP (Natural Language Processing) tasks, and Bert achieved good results [30]. Zhuang et al. proposed the Roberta model in 2021 [31]. The Roberta model added batch size and training data (more than 100 GB) based on the Bert model and used double-byte coding in language representation [32], which improved the accuracy of vocabulary representation and task execution efficiency.

Therefore, based on the above comparative analysis, deep learning models perform well in relation extraction tasks, and we propose a new method of multi-neural network cooperation for relation extraction based on deep learning. The proposed Roberta–BiGRU–FC multi-neural network cooperation model uses Roberta's pre-training model as an encoder and BiGRU and FC model as a decoder, and the model was optimized to better extract entity relations under the condition of a small, balanced dataset.

3. Proposed Method

Aiming at the problem of entity relationship recognition in a small dataset, the overall framework of the RBF model proposed in this paper is illustrated in Figure 1. It mainly includes four levels: (1) the text layer, which refers to the preprocessed text; (2) Roberta layer, using the Roberta pre-training model for text feature coding; (3) the BiGRU layer, decoding text features; and (4) the FC layer, mapping the result to the corresponding relationship category, and outputting the final relationship value according to the rules.



Figure 1. The overall framework of the RBF model.

3.1. Roberta Layer

In the coding layer, in addition to the text that needed to be processed originally, we also inputted the entity pairs as external features and performed feature fusion in the tokenizer stage to improve the effectiveness of the final model. As illustrated in Figure 2, text information was first inputted into Tokenizer Dictionaries, which obtained token vectors based on dictionary-matching methods.

The token embeddings obtained above only represent the characteristic value of a single word at the word level. In addition, other features also have a significant impact on the entity relationships to be extracted, such as text sorting and inter-line order. Therefore, as illustrated in Figure 3, in feature processing it is necessary to integrate features such as position and text line order with the characteristics of the text itself.



Figure 3. Input Embeddings Fusion.

The original text is represented by $x = \{x_1, x_2, ..., x_n\}$. After feature extraction, the calculation methods for the three features were as follows, where W represents the maximum number of characters. In this paper, the threshold of W was set to 128. If each line of text was more than 128 it would be truncated, and if it was less than 128 it would be supplemented with 0. V_k represents the dimension of a character vector.

$$X_{\text{Token}} = R_1^{W \times V_k} \tag{1}$$

$$X_{\text{Position}} = R_2^{W \times V_k} \tag{2}$$

$$X_{\text{Segment}} = R_3^{W \times V_k} \tag{3}$$

The input of the transformer layer is $E = X_{Token} + X_{Position} + X_{Segment}$, where CLS (classification) and SEP (separator) fields are separators specified by Roberta.

After embedding inputs into the Roberta model, the Roberta model continuously extracts sentence features through the multi-head attention mechanism method. Firstly, it multiplies the random initialization matrix M with the fusion feature vector of the above input to obtain vectors Q, vectors K, and vectors V, and then updates and optimizes the values of these three vectors as the training process proceeds. The attention mechanism function represents a mapping from Q to a series of K, and then to V, The expression formula is as follows, and d_k represents the dimension of Q.

Attention(Q, K, V) = softmax(
$$\frac{QK^{T}}{\sqrt{d_{k}}}$$
)V (4)

To prevent the effect of an iteration from being bad, the model performs residual connection and layer normalization after multi-head attention calculation and converts the input vector into data with a mean value of 0 and a variance of 1. The normalization

formula is as follows, where E[x] is the mean value of the sample and $\sigma^2[x]$ is the variance of the sample:

$$y = \frac{x_i - E[x]}{\sqrt{\sigma^2 + \varepsilon}} * r + \beta$$
(5)

The whole process needs to be iterated many times without a segment optimization effect. The structure of the network in a single iteration is illustrated in Figure 4.



Figure 4. Roberta network structure.

3.2. BiGRU Layer

The GRU network is a variant of the LSTM network. It is a variant model of recurrent neural networks that can effectively handle sequential data of text line types. The GRU network converts the forgetting gate and input gate in the LSTM network into an update gate to solve the problem of ladder explosion caused by too long text [33]. The GRU model is simpler and more efficient than the standard LSTM model. Because of this, this paper used the BIGRU model to further decode text feature vectors, hoping to obtain better entity-relationship data features.

This layered network mainly included an update gate and a reset gate. The update gate mainly controlled the retention or deletion of some information from the forward state, and the recharge gate was used to control whether the calculation of the candidate state was dependent on the previous state.

The calculation formula for the updated door is as follows:

$$z_t = \sigma(W_Z \cdot I_t + U_Z P_{t-1}) \tag{6}$$

The calculation formula for reset door is as follows:

$$\mathbf{r}_t = \mathbf{S}(\mathbf{W}_t \cdot \mathbf{I}_t + \mathbf{U}_t \mathbf{P}_{t-1}) \tag{7}$$

The calculation formula of the implicit state is as follows:

$$\widetilde{P}_{t} = \tan h(W \cdot I_{t} + U(r_{t} \otimes P_{t-1}))$$
(8)

The calculation formula of the output state is as follows:

$$P_t = (1 - z_t) \otimes P_{t-1} + z_t \otimes P_t$$
(9)

It represents the features entering the model, h_{t-1} represents the characteristics of the input at time t - 1, and h_t represents the characteristics of the input at time t. S is a sigmoid function, and the value range of S is from 0 to 1.

3.3. Full Connection Layer

At the end of the model, this paper used the FC layer to perform dimensional changes on the output of the model and match the correct size so that the output dimension of the whole network was equal to the number of categories of entity relationship classification, to obtain the optimal entity relationship category. The formula used in the full connection layer is illustrated in Formula (10): m represents the input matrix, A represents the weight matrix, and B represents the paranoid matrix.

$$\mathbf{y} = \mathbf{m}\mathbf{A}^{\mathrm{T}} + \mathbf{b} \tag{10}$$

3.4. Loss Function

The loss function was used to calculate the difference between the predicted result and the real value. The model should use the cross-entropy function to determine the proximity between the actual output and the expected output. Its calculation formula is as follows, where p is the result of the model output and q is the label corresponding to the data.

$$loss(p,q) = -\log(\frac{exp(p[q])}{\sum_{j} exp(p[q])})$$
(11)

3.5. Learning Rate Optimization Based on Cosine Annealing Algorithm

In deep learning technology, the gradient descent algorithm is generally used to optimize the objective function during the operation of the model. When the loss function is at the minimum value, the learning rate should be adjusted to minimize it, to prevent excessive parameter update and overfitting. Especially under the condition of a small balanced sample, the effect is better and more obvious.

The current mainstream methods include the Genetic Algorithm, Particle Swarm Optimization, the Differential Evolution algorithm, the Exponential Decay algorithm, and the Cosine Annealing algorithm. The Genetic Algorithm, Particle Swarm Optimization, Differential Evolution algorithm, and Exponential Decay algorithm are prone to premature convergence and poor convergence ability for high-dimensional complex problems of multi-neural network cooperation [34,35], especially the Particle Swarm Optimization algorithm, which is very prone to premature convergence [36]. Therefore, this paper selected the Cosine Annealing algorithm [37] to optimize the learning rate based on the fusion optimization of the feature layer, and the effectiveness of the method for the results was verified by experiments.

Based on this, the cosine annealing algorithm was adopted to make the learning rate change according to the law of the cosine cycle, In a cosine cycle, the learning rate slows down first and accelerates the decline. In each cycle, the cosine annealing of the learning rate is carried out according to the following formula, where lr_{max} is the highest learning rate initially set, lr_{min} is the lowest learning rate initially set, T_{cur} is the number of iterations epochs since the last restart, and T is the total number of iterations epochs contained in a cycle.

$$lr = lr_{min} + \frac{1}{2}(lr_{max} + lr_{min})\left(1 + \cos\left(\frac{T_{cur}}{T}\right)\pi\right)$$
(12)

This model is the optimal multi-neural network cooperation model found through a large number of experimental processes for small sample conditions. The main differences from existing models are reflected in two aspects: firstly, the various layers in the multi-neural network are different, similar to the existing BBF (BERT–BiGRU–FC) model, but the experimental effect of RBF is stronger than the BBF model. Secondly, there are differences in the optimization methods used. This paper innovatively uses the cosine annealing

algorithm to optimize the multi-neural network cooperation model, and the experimental results show that the optimization effect is effective.

4. Experiments and Results

4.1. Construction of the Small Sample Dataset

DuIE2.0 is a commonly used public dataset for entity relationship extraction at present. Its triple composition mainly comes from Baidu Encyclopedia, Baidu feed flow text, and Baidu tieba. It contains a total of 50 entity relationships, as well as up to 172,983 training sets and 19,981 testing sets. However, there is no average sampling for different relationship categories in DuIE2.0, and there are few small samples of public datasets about entity relationship extraction [38].

To construct the experimental dataset, we analyzed the published DuIE2.0 entity relationship dataset for optimization and simplification. On this basis, we proposed a method that was used to construct a small and balanced dataset. It mainly sets the average value of relationships for big datasets and performs average sampling on the relationships within the big dataset to obtain a simplified average sampling for small datasets. To construct an average sampling small sample dataset, during the experimental process this paper simplified the setting of an average sampling upper limit of 30 for the training set relationship and 10 for the testing set average sampling upper limit. Finally, we built a small sample data set with 462 rows of test data, 1451 rows of training data, and 50 relationship types (including an unknown relationship). Although the dataset was small, there were many types of relationships; this also increased the difficulty of relationship extraction. Some examples of sampling results are illustrated in Table 1.

Relationship Number	DuIE2.0 Train Set	Our Small Balanced Train Set	DuIE2.0 Test Set	Our Small Balanced Test Set	
1	1817	30	171	10	
2	4701	30	460	10	
3	2466	30	237	10	
4	16,553	30	1425	10	
5	10,342	30	889	10	
6	660	30	50	10	
7	1008	30	94	10	
8	3522	30	337	10	
17	20	20	1	1	
	•••				
50	633	30	57	10	
Total number	172,983	1451	19,981	462	

Table 1. Example of dataset sampling results.

4.2. Evaluation Index

For the evaluation of experimental results, this paper takes the F1 value as the main evaluation index based on current mainstream practices. P is the precision rate of experimental results, R is the recall rate of experimental results, FP is the number of times predicted to be the relationship but not the relationship, TP is the number of correct predictions of the relationship, FN is the number of times predicted to be the relationship but wrong, and the specific calculation formula of F1 value is as follows:

$$P = \frac{TP}{TP + FP}$$
(13)

$$R = \frac{TP}{TP + FN}$$
(14)

$$F1 = \frac{2P * R}{P + R}$$
(15)

4.3. Baselines

In terms of multi-neural network cooperation for the relation extraction task, we compared our model with the following models:

- In 2015, Shu Zhang, etc., proposed using bidirectional long short-term memory networks for relation classification [39].
- In 2018, Zhao Ming proposed a Word2vec–BIGRU–FC model for relationship extraction [40].
- In 2020, Zihan Wang, etc., proposed a Bert–BiLSTM–FC model for relationship extraction [41].
- In 2021, YUE Qi and Li Xiang proposed a Bert–RNN–FC model for relationship extraction [42].
- In 2022, Gupta, etc., proposed a Bert–BiGRU–FC model for relationship extraction [43].
 Because there are few existing models for relation extraction under the condition of

Chinese small samples, and we were unable to find more mature baseline models, this paper also used the variant of the RBF model (before optimization) as a baseline and the variant of the RBF model using the Exponential Decay algorithm as a baseline.

4.4. Parameter Setting

The server configuration used in this experiment was Intel (R) Core (TM) i7-10875H (2.3 GHz), with 32 GB memory, NVIDIA Geforce RTX 2060 GPU, and a Windows 11 64-bit operating system. The version of the PyTorch framework the experiment used was 1.7.1. The version of Word2vec used was Tencent-ailab-embedding-zh-d100-v0.2.0-s from Tencent AILab. The experiment also used the transformers model, whose version number was 2.5.1, and the coding layer used the Roberta model and the BiGRU model.

The main parameter settings of the Roberta are illustrated in Table 2.

Parameter	Value	
hidden_size	768	
max_position_embeddings	512	
num_attention_heads	12	
num_hidden_layers	12	
pooler_fc_size	768	
pooler_num_attention_heads	12	
pooler_num_fc_layers	3	
pooler_size_per_head	128	
vocab_size	21,128	

Table 2. The parameter settings of the Roberta model.

The main parameter settings of the BiGRU model are illustrated in Table 3.

Table 3. The BiGRU model parameter settings.

Parameter	Value
input_size	768
hidden_size	64
dropout	0.4

The learning rate was optimized based on cosine annealing. According to Formula (11), the initial maximum learning rate was set to 0.01 and the cycle period of the learning rate was set to 5 epochs. The optimization of the learning rate in our method is illustrated in Figure 5.



Figure 5. Learning rate optimization based on cosine annealing.

4.5. Main Results and Analysis

In terms of experimental comparison, on the small sample corpus, this method was compared with two types of current commonly used entity classification methods. In the feature extraction layer, experimental comparisons were conducted on the traditional Word2vec model and Bert model, and the Word2vec model used version 0.2.0 "Tencent AI Lab embedding Corps" published by Tencent AI laboratory. The Bert model used the Roberta model. The parameter settings and results obtained are illustrated in Table 4.

Table 4.	Comparison	of experime	ntal results.
lable 4.	Comparison	of experime	ntal results.

No.	Model	Training Set	Precision	Recall	F1	Best Epoch
1	Word2vec-GRU-FC	Epoch = 145	44.4%	41.5%	42.6%	138
2	Word2vec-BiGRU-FC (Zhao Ming et al., 2018 [40])	learning_rate = 0.002, embedding_dim = 100,	65.4%	66.6%	66.0%	46
3	Word2vec-LSTM-FC	hidden_dim = 64, BATCH = 4, dropout = 0.4	23.6%	32.9%	27.5%	139
4	Word2vec-BiLSTM-FC (Shu Zhang et al., 2015 [39])		58.1%	59.2%	58.6%	131
5	Bert–BiGRU–FC (Gupta et al., 2022 [43])		75.6%	74.4%	76.0%	134
6	Bert-BiLSTM-FC (Zihan Wang and Bo Yang, 2020 [41])		74.6%	72.0%	73.3%	138
7	Bert–RNN–FC (YUE Qi and Li Xiang, 2021 [42])		73.5%	72.4%	72.9%	89
8	RBF before optimization	Epoch = 145, learning_rate = 0.004 dim = 100, hidden_dim = 64, BATCH = 4, dropout = 0.4	89.8%	89.4%	89.6%	34
9	RBF using Exponential Decay algorithm	Epoch = 145, learning_rate using the Exponential Decay algorithm, embedding_dim = 100, hidden_dim = 64, BATCH = 4, dropout = 0.4	90.8%	90.2%	90.5%	47
10	Our RBF model	Epoch = 145, learning_rate using cosine annealing, embedding_dim = 100, hidden_dim = 64, BATCH = 4, dropout = 0.4	91.0%	91.6%	91.9%	47

Due to the small amount of data, the times required for completing 145 epochs in columns 8, 9, and 10 with better training results in Table 4 were 1:31:02, 1:36:27, and 1:27:40. There was not much difference between them. The F1 and the loss value of other models on multi-neural network cooperation under different iteration times are illustrated in Figures 6 and 7. They mainly included two categories: using Word2vec as the encoder, and using Bert as the encoder.



Figure 6. Relationship extraction effect of models with Word2vec as the encoder. (**a**) The variation in F1 value at different epochs; (**b**) the variation in LOSS value at different epochs.



Figure 7. Relationship extraction effect of models with Bert as the encoder. (**a**) The variation in F1 value at different epochs; (**b**) the variation in LOSS value at different epochs.

The relationship extraction effect of our RBF model before and after optimization is illustrated in Figure 8.

Further experiments were conducted to optimize the model. For the RBF model, the cosine annealing algorithm was compared with the exponential decay algorithm, and the results are shown in Figure 9.



Figure 8. Relationship extraction effect of our RBF model before and after optimization. (**a**) The variation in F1 value at different epochs; (**b**) the variation in LOSS value at different epochs.



Figure 9. Relationship extraction effect of our RBF model using different optimization algorithms. (a) Learning rate optimization based on the Exponential Decay algorithm; (b) the variation in F1 value at different epochs.

5. Discussion

It can be seen from Table 4 that there was little difference between the precision rate, recall rate, and F1 value, indicating that there were no overfitting phenomena in this experiment, indicating the effectiveness of this experiment.

As can be seen from the Table 4 results, in the task of extracting entity relations from small samples, the precision rate of the proposed model was 91.0%, the recall rate was 91.6%, and the F1 value was 91.9%. The F1 value of the proposed model is 25.9% higher than the Word2vec–BiGRU–FC model and 33.1% higher than the Word2vec–Bilstm–FC model. For the current mainstream models, compared with Bert–BiCNN–RNN, the F1 value increased by 19%; compared with Bert–BiLSTM–FC, the F1 value increased by 18.6%; and compared with Bert–BiGRU–FC, the F1 value increased by 15.9%. It can be seen from the above results that the method proposed in this paper was greatly improved compared with other methods under the condition of small samples.

It can be seen in Figure 6 that when the number of epochs increased to 50, the F1 value tended to be stable at around 60% with Word2vec as the encoder. It can be seen in Figure 7 that when the number of epochs increased to 30, the F1 value tended to be stable around

70% with Bert as the encoder. It can be concluded that under the condition of small samples, the effect of using Bert as an encoder is better than using traditional Word2vec; this may be due to the strong fitting ability of Bert. In addition, using BiGRU as a decoder is better than using other models as decoder; it shows the excellent decoding ability of the BiGRU model, but the effect is still not ideal. Given this, we put forward the Roberta–BiGRU–FC model and optimized the feature fusion of the input layer and the learning rate of the model.

It can be seen in Figure 8 that the optimized model had a good effect in the first round of epochs. The effect of the original model increased rapidly in the early stage; this may be due to the stable learning rate of the original model, but as the number of epochs increased to 30, the effect of the optimized model became better than the original model and tended to be stable around 91%. By comparing with the exponential decay algorithm in Figure 9, the result showed that the cosine annealing algorithm could better converge the proposed multi-neural network cooperation model and obtain better results, and the optimization of the feature input layer and learning rate could improve the training effect of the RBF model; our optimization of the RBF model was effective with a small data set.

According to the above experimental results, it can be seen that, first, the small balanced dataset constructed in this paper could support the detection of the model; second, compared with other models, the RBF model proposed in this paper could achieve better results in the task of extracting entity relations from the small balanced dataset; third, we had a certain effect on the optimization of RBF model.

6. Error Analysis

From the experimental results in Table 4 and Figure 8, it can be seen that although the optimal F1 value of the RBF model was higher than other models, there was, however, a lack of stability in the image, especially after the use of the cosine annealing algorithm. There was significant shaking in the first 40 epochs, one shaking episode around the 80th epoch, and one shaking episode around the 130th epoch. Therefore, the drawback of this model is that it is prone to significant F1 value jitter during the training process.

7. Conclusions

Aiming at the task of entity relationship extraction on a small balanced sample dataset, this paper proposed a multi-neural network collaborative RBF model. The model was also optimized for learning rate by cosine annealing. In addition, this paper constructed a relatively uniformly sampled small-sample dataset. Experiments showed that compared with other models, this method could achieve good results in entity relationship extraction tasks. In subsequent research, we will conduct more comparative experiments on optimizing learning rates, such as introducing genetic algorithms into experiments and conducting comparisons, and we will make more efforts in the training effect of the model, such as introducing the attention mechanism into the decoding layer of the model and improving the performance of the model to a certain extent.

Author Contributions: Y.L.: conceptualization, writing—original draft, writing—reviewing and editing, methodology, validation, formal analysis, data curation. Q.Z.: supervision, writing—reviewing and editing. X.W.: supervision, formal analysis. T.Z.: supervision, writing—reviewing and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset and code generated during the current study are not publicly available, because the data and code also form part of the ongoing study, but they can be obtained from the corresponding authors according to reasonable requirements.

Acknowledgments: The authors thank all reviewers who helped to improve this manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

- 1. Ardagna, D.; Barbierato, E.; Gianniti, E.; Gribaudo, M.; Pinto, T.B.; da Silva, A.P.C.; Almeida, J.M. Predicting the performance of big data applications on the cloud. *J. Supercomput.* **2021**, *77*, 1321–1353. [CrossRef]
- 2. Zhao, M.; Zhao, Y.; Zhao, Y.; Luo, G. A new joint model for extracting overlapping relations based on deeplearning. J. Univ. Chin. Acad. Sci. 2022, 39, 240.
- 3. Wu, H.; Lu, L.; Yu, B. Chinese named entity recognition based on transfer learning and bilstm-crf. J. Chin. Comput. Syst. 2019, 40, 1142–1147.
- 4. Qiao, B.; Zou, Z.; Huang, Y.; Fang, K.; Zhu, X.; Chen, Y. A joint model for entity and relation extraction based on BERT. *Neural Comput. Appl.* **2022**, *34*, 2739–2748. [CrossRef]
- 5. Ma, L.; Ren, H.; Zhang, X. Effective cascade dual-decoder model for joint entity and relation extraction. *arXiv* 2021, arXiv:2106.14163.
- Zhang, L.; Zhao, H. Named entity recognition for Chinese microblog with convolutional neural network. In Proceedings of the 2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Guilin, China, 29–31 July 2017; pp. 87–92.
- 7. Li, D.M.; Zhang, Y.; Li, D.Y.; Lin, D.Q. Review of entity relation extraction methods. J. Comput. Res. Dev. 2020, 57, 1424–1448.
- E, H.H.; Zhang, W.J.; Xiao, S.Q.; Cheng, R.; Hu, Y.X.; Zhou, X.S.; Niu, P.Q. Survey of entity relationship extraction based on deeplearning. J. Softw. 2019, 30, 1793–1818. [CrossRef]
- 9. Rokach, L. Ensemble-based classifiers. Artif. Intell. Rev. 2010, 33, 1–39. [CrossRef]
- 10. Zhang, Q.; Wu, M.; Lv, P.; Zhang, M.; Lv, L. Research on Chinese Medical Entity Relation Extraction Based on Syntactic Dependency Structure Information. *Appl. Sci.* **2022**, *12*, 9781. [CrossRef]
- 11. Aone, C.; Halverson, L.; Hampton, T.; Ramos-Santacruz, M. SRA: Description of the IE2 System Used for MUC-7. In Proceedings of the 7th Message Understanding Conference, Fairfax, VA, USA, 29 April–1 May 1998.
- 12. Zelenko, D.; Aone, C.; Richardella, A. Kernel methods for relation extraction. J. Mach. Learn. Res. 2003, 3, 71–78.
- 13. Fundel, K.; Küffner, R.; Zimmer, R. RelEx—Relation extraction using dependency parse trees. *Bioinformatics* 2007, 23, 365–371. [CrossRef] [PubMed]
- 14. Zhu, J.; Nie, Z.; Liu, X.; Zhang, B.; Wen, J.R. StatSnowball: A statistical approach to extracting entity relationships. In Proceedings of the 18th International Conference on World Wide Web, Madrid, Spain, 20–24 April 2009; pp. 101–110.
- 15. Zhou, G.; Sun, J.; Zhang, J.; Zhang, M. Exploring Various Knowledge in Relation Extraction. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, Ann Arbor, MI, USA, 25–30 June 2005; pp. 427–434.
- Culotta, A.; McCallum, A.; Betz, J. Integrating probabilistic extraction models and data mining to discober relations and patterns intext. In Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, New York, NY, USA, 4–9 June 2006; pp. 296–303.
- Li, L.; Jing, L.; Huang, D. Protein-protein interaction extraction from biomedical literatures based on modified SVM-KNN. In Proceedings of the 2009 International Conference on Natural Language Processing and Knowledge Engineering, Dalian, China, 24–27 September 2009; pp. 1–7.
- 18. Wang, C.D.; Xu, J.; Zhang, Y. Summary of entity relationship extraction. Comput. Eng. Appl. 2020, 56, 25–36.
- 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.
- 20. Santos, C.N.D.; Xiang, B.; Zhou, B. Classifying relations by ranking with convolutional neural networks. *arXiv* 2015, arXiv:1504.06580.
- 21. Zhou, X.; Liu, L.; Luo, X.; Chen, H.; Qing, L.; He, X. Joint Entity and Relation Extraction Based on Reinforcement Learning. *IEEE Access* 2019, *7*, 2169–3536. [CrossRef]
- 22. Liu, C.; Sun, W.; Chao, W.; Che, W. Convolution neural network for relation extraction. In *International Conference on Advanced Data Mining and Applications*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 231–242.
- 23. Zhang, D.; Wang, D. Relation classification via recurrent neural network. *arXiv* 2015, arXiv:1508.01006.
- 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. pp. 207–212.
- 25. Bengio, Y.; Ducharme, R.; Vincent, P. A neural probabilistic language model. Adv. Neural Inform. Proc. Syst. 2000, 13.
- 26. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed resentations of word and phrases and their compositonality. *Adv. Neural Inf. Process. Syst.* 2013, 26, 3111–3119.
- 27. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *Comput. Lang.* **2018**, *23*, 3–19.
- 28. Shi, P.; Lin, J. Simple bert models for relation extraction and semantic role labeling. *arXiv* **2019**, arXiv:1904.05255.
- 29. Wan, C.X.; Li, B. Financial causal sentence recognition based on BERT-CNN text classification. J. Supercomput. 2022, 78, 6503–6527. [CrossRef]

- Sun, C.; Qiu, X.; Xu, Y.; Huang, X. How to fine-tune bert for text classification. In *China National Conference on Chinese Computational* Linguistics; Springer: Cham, Switzerland, 2019; pp. 194–206.
- Zhuang, L.; Wayne, L.; Ya, S.; Jun, Z. A robustly optimized BERT pre-training approach with post-training. In Proceedings of the 20th Chinese National Conference on Computational Linguistics, Huhhot, China, 10–12 August 2021; pp. 1218–1227.
- Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. arXiv 2019, arXiv:1907.11692.
- 33. Bai, T.; Wang, C.; Wang, Y.; Huang, L.; Xing, F. A novel deep learning method for extracting unspecific biomedical relation. *Concurr. Comput. Pract. Exp.* **2020**, *32*, e5005. [CrossRef]
- Goudarzi, S.; Hassan, W.H.; Anisi, M.H.; Soleymani, S.A. Comparison between hybridized algorithm of GA–SA and ABC, GA, DE and PSO for vertical-handover in heterogeneous wireless networks. *Sādhanā* 2016, 41, 727–753. [CrossRef]
- 35. Ouyang, P.; Pano, V. Comparative study of DE, PSO and GA for position domain PID controller tuning. *Algorithms* **2015**, *8*, 697–711. [CrossRef]
- 36. Yarat, S.; Senan, S.; Orman, Z. A Comparative Study on PSO with Other Metaheuristic Methods. *Appl. Part. Swarm Optim. New Solut. Cases Optim. Portf.* **2021**, *31*, 49–72.
- Jouhari, H.; Lei, D.; Al-Qaness, M.A.A.; Elaziz, M.A.; Ewees, A.A.; Farouk, O. Sine-cosine algorithm to enhance simulated annealing for unrelated parallel machine scheduling with setup times. *Mathematics* 2019, 7, 1120. [CrossRef]
- Wang, Y.; Zhou, Z.; Jin, S.; Liu, D.; Lu, M. Comparisons and selections of features and classifiers for short text classification. *IOP Conf. Ser. Mater. Sci. Eng.* 2017, 261, 012018. [CrossRef]
- 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.
- Zhao, M.; Dong, C.; Dong, Q.; Chen, Y. Question Classification of Tomato Pests and Diseases Question Answering System Based on BIGRU. *Trans. Chin. Soc. Agric. Mach.* 2018, 49, 271–276.
- Wang, Z.; Yang, B. Attention-based bidirectional long short-term memory networks for relation classification using knowledge distillation from BERT. In Proceedings of the 2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech), Calgary, AB, Canada, 17–22 August 2020; pp. 562–568.
- Yue, Q.; Li, X. Research on the Construction of Chinese Forestry Knowledge Graph Based on BERT and Bi-directional RNN. J. Inn. Mong. Univ. 2021, 52, 1000–1638.
- 43. Wang, H.; Wang, X.; Miao, Y.; Xu, T.; Liu, Z.; Wu, H. Densely Connected BiGRU Neural Network Based on BERT and Attention Mechanism for Chinese Agriculture-related Question Similarity Matching. *Trans. Chin. Soc. Agric. Mach.* **2022**, *53*, 244–252.

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.