IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 26, 2020, accepted June 11, 2020, date of publication June 17, 2020, date of current version July 29, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.3002863*

# BERT-Based Chinese Relation Extraction for Public Security

**JIAQI HOU[1], XIN LI[1], HAIPENG YAO[2], (Senior Member, IEEE), HAICHUN SUN[1], TIANLE MAI[2], (Graduate Student Member, IEEE), AND RONGCHEN ZHU[1]**

[1]School of Information Technology and Cyber Security, People's Public Security University of China, Beijing 100038, China
[2]School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China

Corresponding author: Xin Li (lixin@ppsuc.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFC0803700, in part by the People's Public Security University of China 2019 Basic Research Operating Expenses New Teacher Research Startup Fund Project under Grant 2019JKF424, and in part by the Natural Science Foundation of China under Grant 41971367.

**ABSTRACT** The past few years have witnessed some public safety incidents occurring around the world. With the advent of the big data era, effectively extracting public security information from the internet has become of great significance. Up to hundreds of TBs of data are injected into the network every second, and thus it is impossible to process them manually. Natural Language Processing (NLP) is dedicated to the development of an intelligent system for effective text information mining. By analysing the text and quickly extracting the relationships between the relevant entities, NLP can establish the knowledge graph (KG) of public security, which lays the foundation for safety case analysis, information monitoring, and activity tracking and locating. One of the current pre-training relation extraction models is the Word2Vec model. The Word2vec model is single mapped, and it produces a static, single representation of the words in sentences. Then, the BERT model considers contextual information and provides more dynamic, richer vector representations of generated words. Therefore, in this paper, we propose a Bidirectional Encoder Representation from Transformers (BERT) based on the Chinese relation extraction algorithm for public security, which can effectively mine security information. The BERT model is obtained by training the Masked Language Model and predicting the next sentence task, which is based on the Transformer Encoder and the main model structure is the stacked Transformers. Extensive simulations are conducted to evaluate our proposed algorithm in comparison to some state-of-the-art schemes.

**INDEX TERMS** BERT, relationship extraction, public security, MaxPooler, ReLU.

## I. INTRODUCTION

In recent years, public safety incidents around the world occur from time to time. Extreme terrorism cases, such as the explosions at Manchester stadium and the St Petersburg subway, pose great challenges to national security services [1]. With the development of the internet and communication technology, effective web text disposal and understanding plays an important role in today's public safety [2].

However, with the massive increase of in the amount of data, where hundreds of TBs of data are produced on the internet every second, manual processing becomes difficult. Recently, NLP has been dedicated to the development of a computer system for effective web text processing [3]. NLP can improve the task efficiency of public security information extraction, classification, and prediction. The traditional human annotation method cannot effectively achieve deep semantic understanding. At present, structured knowledge such as Wikidata, Yago and DBpedia have been widely used in NLP applications. [4]–[7]. Because of the large scale of knowledge and the high costs of manual tagging, the technology of relationship extraction technology becomes particularly important since it can add more abundant world knowledge to the KGs and automatically obtain world knowledge [8].

Specifically, given the sentence and entities, the entity relation extraction model needs to determine the relationship between entities according to the semantic information. For example, given the sentence ''Tsinghua university is located near Beijing'' and the entities ''Tsinghua university''

The associate editor coordinating the review of this manuscript and approving it for publication was Keli Xiao.

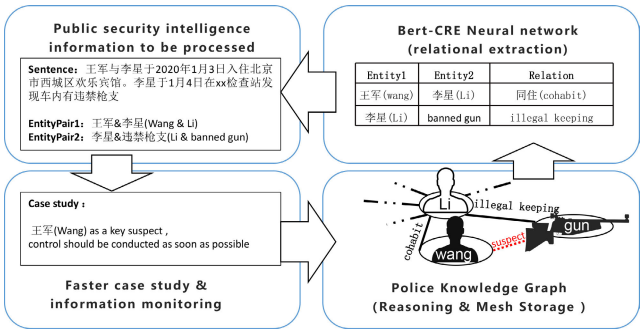



**FIGURE 1.** Public safety scene sample.

and ''Beijing'', the model can semantically obtain the ''located'' relationship, and finally extract the ternary knowledge (Tsinghua University, located in, Beijing). Because the core of Natural Language Understanding(NLU) is to establish a KG of a field, and the relation extraction is the core of KG construction [9].

However, because of the nature of Chinese text, Chinese relation extraction is still at the initial stage.

The current pre-training relation extraction model is the Word2Vec model. The word representations it produces are static, they do not consider the context, and they cannot solve the problem of polysemy. Meanwhile, the BERT model uses the Transformer encoder as the feature extractor. This method naturally makes good use of the context, dynamically produces word vector representations, and trains MLM such as denoising targets on large-scale corpora. The resulting representation is very helpful for downstream tasks and solves the polysemy problem [10], [11].

BERT can reflect the complex characteristics of words, including their syntax and semantics. The word embedding method such as word2vec does not have this advantage itself because it is too simple. The BERT model learns a ''deep'' network; therefore, after the pre-training, it can get different levels of features using different network layers. Generally, the high-level generated features better reflect the abstract and context-dependent part while the bottom generated features pay more attention to the grammatical part.

Therefore, in this paper, we proposed a BERT Chinese Relational Extraction architecture(Bert-CRE). The architecture consists of an input layer, Embedding layer, Encoder layer, Pooling layer, and fully connected layer. The input layer first receives the entity pair features, the original sentence, the relation, the entity pair position and the sentence coding. Using the BERT Chinese-based preprocessing model, the following embedding, encoding, and pooling operations are performed on the data. In addition, the entity-to-position matrix processed by fine-tuning is the input of the fully connected layer. At last, the training set is used for 20 epochs of training, and the verification set is used for evaluation [12].

Compared with the 100-dimensional Word2Vec model, the BERT model can reach 768-dimensional [13]. BERT-Chinese-Based is trained with a large number of Chinese corpora to express more semantic information. BERT-Chinese-Based can effectively handle the polysemous words. It is able to make full use of the position information of entity pairs, improve the recognition degree of entity relations, emphatically understand semantics, and analyse sentence semantics. Our relationship extraction model can be applied to customs and exports, hazardous chemical content control and investigation, and a variety of security scenarios.

Here, the contributions of this article can be summarized as follows. (1) In RELATED WORK, we present the related work on relation extraction methods. (2) In Section A of the METHODOLOGY, we introduce the Transformer Encoder into the BERT-based pre-training model using the Scaled Dot-Product attention mechanism in the task of relation extraction. The use of the BERT-based pre-training model helps to solve the problem of the polysemy of a word and capture more semantic features between words. (3) In Section B of the METHODOLOGY, we propose a variety of downstream network processing layers, including a position embedding layer, dropout layer and max pooling linear layer, further enhances the extraction effect. (4) Compared with the current best model, the proposed model can obtain more precise representations of relations and achieve better performance. To demonstrate the advantages of the proposed model, we compare it to other relation extraction models with different downstream network processing layers and illustrate the experiment results in SIMULATION EXPERIMENTS AND RESULTS ANALYSIS. Finally, the conclusion is given.

## II. RELATED WORK

Regarding neural network relationship extraction, in [14], Liu *et al*. used Convolutional Neural Network (CNN) in relational extraction for the first time. The advantage of this paper is that it introduces the CNN structure to relational extraction. The disadvantage is that the CNN has a relatively simple structure, and there is no Pooling layer, which may make the effects of noise more obvious. In [15], Zeng *et al*. proposed to extract the optimal features corresponding to each convolution kernel and introduced position features. The drawback is that while multiple convolution kernels are used, only the same window size is used. In [16], Santos *et al*. used the advantage of the ranking loss, which has improved the results by more than 2 percent. In [17], Zhang *et al*. used a Recurrent Neural Network (RNN), which has a similar effect as a CNN. During long text modelling, the memory advantage of the RNN can be seen. In [18], Zhou *et al*. used the NLP task's standardized Attention Bi-directional Long Short-Term Memory (BiLSTM), which also worked well on relation classification. In [19], Huh *et al*. proposed an Attention model that reach the highest F1-Score of 88 percent on the SemEval 2010 Task8 dataset and was not surpassed until 2019. The advantage is that it uses two-tier Attention. In [20], Ren *et al*. used an external message, the Entity Description, to address the data sparsity problem in the relational classification task. This paper proposes

**TABLE 1.** The Full form of the abbreviation in the text.

| Full form | Abbreviation |
|---|---|
| Bidirectional Encoder Representation from Transformers | BERT |
| Natural Language Processing | NLP |
| knowledge graph | KG |
| Natural Language Understanding | NLU |
| Bert Chinese Relational Extraction architecture | Bert-CRE |
| Convolutional Neural Networks | CNN |
| Recurrent Neural Networks | RNN |
| Bi-directional Long Short-Term Memory | BiLSTM |
| Neural Relation Classification with Text Descriptions | DesRC |
| Generative Pre-Training | GPT |
| Embeddings from Language Model | ELMo |
| Rectifiedlinearunit | ReLU |
| Multi-Grained Lattice | MG-lattice |
| Area Under the Curve | AUC |
| Precision-Recall plot | P-R plot |
| Continuous Bag-of-Word Model | CBOW |
| Long short-term memory | LSTM |

a Neural Relation Classification with Text Descriptions (DesR, neural relation classification method), which describes the entity's text options and adds them to the neural network model as supplementary information.

At present, RNNs and their various improved algorithms are the mostly used in relation extraction tasks, but none of these methods take into account the dependence between the relationships in sentences and do not fully utilize the indirect relationship between entities when dealing with the existence of multiple entity pairs in sentences. In [21], Fenia *et al*. proposed implicit inferences that, unlike these methods that existed in 2018, used a path-based entity graph model, which makes full use of the indirect relationships between entities when identifying the relationships of the target entity pairs. Although, some other algorithms also address the multiple relationships in sentences. However, these algorithms cannot model known entity paths. The feasibility and effectiveness of the path-based graph model in relational extraction are proved by experiments. By 2019 the NLP field began many attempts to improve the preprocessing model, mainly using the BERT, Generative Pre-Training (GPT), and the Embeddings from Language Model (ELMo) model. Among them, in [22] Alt *et al*. the GPT was introduced for pre-training. In [21], for the 2019 ACL, Christopoulou *et al*. proposed general-purpose relation extraction, using BERT for relational representation, and proposed the Matching the Blanks pre-training task. The paper's model achieves SOTA results on multiple datasets, and the promotion is obvious in the case of small samples.

## III. METHODOLOGY

Our architecture contains a BERT model layer, a Dropout layer and a Classification layer, where the BERT model layer effectively implements the embedding, Transformer encoding, and pooling. Through the Embedded, Encoder and Pooling layers, it output the feature vectors of each sentence. Based on this, the entity-to-location information features are embedded by the feature vectors of each sentence. Then, the dropout layer is applied to prevent overfitting. Eventually, all

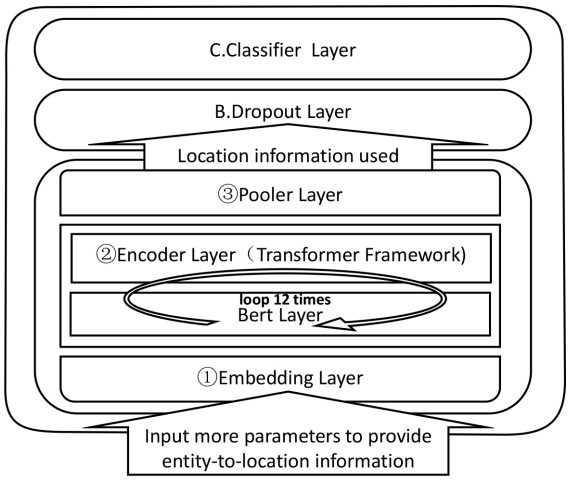



**FIGURE 2.** Model structure framework.

the model weights are trained, the entity-to-relationship prediction of each sentence is realized and the total loss is obtained.

1. The input layer first receives the entity pair features, the original sentence, the relation, the entity pair position and the sentence coding.

2. The features are transformed into 768-dimensional word & position & token vectors in the embedding layer. The three kinds of embeddings are added together and the last output is returned.

3. The BERT Encoding layer establishes the entire Transformer architecture, which contains several layers, the core of which is the Scaled Dot-Product Attention structure.

4. The output of the encoder layer is fed into the pooling layer and pooled.

5. Through fine-tuning, the entity-to-position matrix enters into the fully connected layer. The output is changed from the token level to the sentence level (sum or max pooling here), and then after dropout are the layerNorm, Rectified Linear Unit (ReLU) and Linear operations. The training set is used for 20 epochs of training and the validation set is used to evaluate.

### A. BERT-BASED PRE-TRAINING MODEL

The BERT model is to process the training set through an attention mechanism. Then, the pre-trained word vectors are loaded through the Embedding layer and the Encoder layer. Finally, the Pooling layer uses the BERT model to train two sentences.

#### 1) BERTEMBEDDING

In the Embedding layer, the input data first passes through the BERT embedding section [23]. It changes each word into $\text{embedding}_{\text{word}}$, $\text{embedding}_{\text{position}}$ and $\text{embedding}_{\text{token\_type}}$ (formula 1).

$$\text{E} = \text{embedding}_{\text{word}} + \text{embedding}_{\text{position}} + \text{embedding}_{\text{token\_type}}. \quad (1)$$

From the papers on BERT, we know that BERT's word vectors are mainly composed of three vectors, which are the vectors of the word itself embedding$_{\text{word}}$, the position of the word in the sentence embedding$_{\text{position}}$ and the position of the sentence in the individual training text embedding$_{\text{token_type}}$ [24]. The advantage is that the BERT based embedding model prediction can handle polysemous words. When the three vectors are added together, after normalization and dropout processing, they will be input into the BERT Encoder model. All above matrices are $d_{batch\_size} \times d_{\max\_length} \times d_{hidden\_\dim}$.

### 2) BERT ENCODER

BERT uses a Transformer Encoder as a language model, and the Transformer model adopts the Attention mechanism to compute the relationship between the input and output. Then, it uses the attention mechanism to compute the relationship between the input-output [25].

The difference between the BERT model and the OpenAI GPT is that the Transformer Encoder is applied, that is, the attention calculation of each moment can get the input of the whole moment before and after [26]. Meanwhile, the OpenAI GPT adopts the Transformer Encoder in which the attention calculation of each moment can only depend on the input of all the moments before that moment.

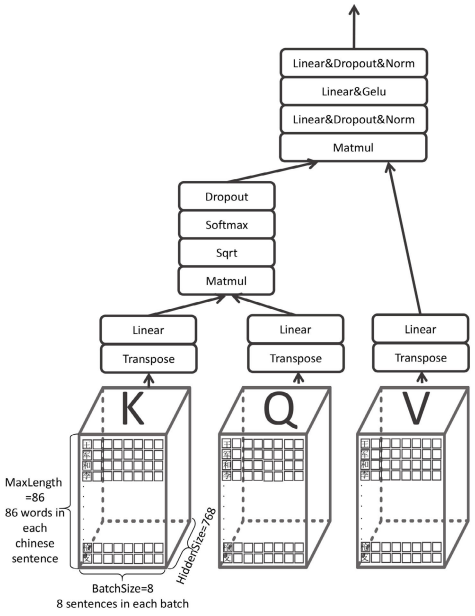



**FIGURE 3.** Scaled dot-product attention structure.

First, $Q$, $K$, and $V$ are three embedding matrices that be changed through the linear processing, corresponding to entering the Multi-Head Attention. The Multi-Head Attention consists of overlapping Scaled Dot-Product Attention. The Scaled Dot-Product Attention structure inside the Multi-Head Attention is shown below (Figure 3).

The input contains the $d_{batch\_size} \times d_{\max\_length} \times d_k$ dimensions $Q$ and $K$, as well as the $d_{batch\_size} \times d_{\max\_length} \times d_v$ dimension $V$. In formula 2, we calculate the dot product of $Q$ and each individual $K$, and divide it by the normalized $\sqrt{d_k}$. Then, we use the Softmax activation to generate the weights. The advantages of the dot product attention mechanism are its low time and space complexity.

$$sim_{Q,K} = \frac{QK^T}{\sqrt{d_k}}. \tag{2}$$

$$sim_{scale}(sim_{Q,K,i}) = \frac{\exp(sim_{Q,K,i})}{\sum_j \exp(sim_{Q,K,j})}. \tag{3}$$

$$Attention(Q, K, V) = sim_{scale} \cdot V. \tag{4}$$

Next, we apply the Softmax function to an n-dimensional input Tensor, rescaling them so that the elements of the n-dimensionalz output Tensor lie in the range $[0, 1]$ and sum to 1. The above is the process of calculating the Scaled Dot-Product Attention.

The $d_{batch\_size} \times d_{\max\_length} \times d_{hidden\_\dim}$ dimensions $K$, $V$ and $Q$ are mapped into dimension $d_{batch\_size} \times d_{\max\_length} \times d_k$, dimension $d_{batch\_size} \times d_{\max\_length} \times d_k$ and dimension $d_{batch\_size} \times d_{\max\_length} \times d_v$, respectively, using different linear transformations of h. Then the attention mechanism is substituted to produce a dimension of $d_{batch\_size} \times d_{\max\_length} \times h \times d_{v\_h}$ (formula 5) output, and then they are put together (formula 6) to obtain the final output with a linear transformation $W^0$.

$$\text{head}_i = Attention(QW_i^Q, KW_i^K, VW_i^V). \tag{5}$$

$$MultiHead(Q, K, V) = Concat(head_1, \ldots, head_h)W^O. \tag{6}$$

Then, we conduct linear processing (formula 7), a dropout and the layerNorm (formula 8). The Multi-Head Attention is done.

$$MH(Q, K, V) = MultiHead(Q, K, V)A^T + b. \tag{7}$$

$$MH(Q, K, V) = \frac{x - E[MH(Q, K, V)]}{\sqrt{Var[MultiHead(Q, K, V)] + \varepsilon}} * \gamma + \beta. \tag{8}$$

Afterward, take another round of linear processing (formula 9) and an activation function (formula 10) are applied, followed by a dropout and a normalization. Thus, the entire Transformer section is complete.

$$M = MH(Q, K, V)A_{M\_Head}{}^T + b_{M\_head}. \tag{9}$$

Apply the Gaussian Error Linear Units function as follows:

$$M = GULE(M) = M * \Phi(M). \tag{10}$$

where is the Cumulative Distribution Function for a Gaussian Distribution.

### 3) BertPooler

This is an activation function that conducts linear processing and uses $Tanh()$ to pool the output of the BERT Encoder. Linearly transform the input data as follows:

$$P = MA_{pool}{}^T + b_{pool}. \tag{11}$$

Then, apply the element-wise function:

$$\underset{\text{token}}{\mathrm{W}} = \tanh(P) = \frac{e^{P} - e^{-P}}{e^{P} + e^{-P}}. \tag{12}$$

### B. POSITION EMB DROPOUT MAX POOLING, AND LINEAR

The best-verified setting for the Learning Rate is 3e-5

#### 1) POSITION EMB

After the sentence is extracted with token level features, the entity pair has embedded position information in the form of a matrix. Using the maximum sentence length (here it is set to 86, that is, the maximum length of a sentence is 86 words) as the length and width, the unit matrix is generated, and the position information is placed on the diagonal line. The following is used:

$$\mathrm{W}_{entity-pos} = \begin{bmatrix} 000000.....0 \\ 010000.....0 \\ 001000.....0 \\ 000000.....0 \\ 000020.....0 \\ 000002.....0 \\ 000000.....0 \\ ..... \\ 000000.....0 \end{bmatrix} \quad \text{Sentence：让小华和小明…}$$

**FIGURE 4.** Position matrix.

As shown in Figure 4, ***Xiao Hua*** and ***Xiao Ming*** are entities 1 and 2, respectively. Through the entity-to-position matrix, we can locate the positions of the entities. We then multiply the feature vector matrix of the sentence with the newly generated entity-to-position matrix. Then we can get the sentence feature vector with the entity pair position information.

$$\underset{\text{token}}{\mathrm{W}} = \underset{\text{token}}{\mathrm{W}} \cdot \underset{\text{entity-pos}}{\mathrm{W}}. \tag{13}$$

The dimensions of all three matrices are $d_{batch\_size} \times d_{\max\_length} \times d_{hidden\_\dim}$

#### 2) SUM AND DROPOUT

In the word dimension, all embeddings are summed, that is, $d_{batch\_size} \times d_{\max\_length} \times d_{hidden\_\dim}$ converted into $d_{batch\_size} \times d_{hidden\_\dim}$, and the sentence feature vector matrix is transferred from the token level to the sentence level. Furthermore, dropout prevents overfitting.

#### 3) MAX POOLING AND DROPOUT

Apply a 1D max pooling over an input signal composed of several input planes. In the simplest case, the output value of the layer with the set input size $B \times M \times H = d_{batch\_size} \times d_{\max\_length} \times d_{hidden\_\dim}$ and output $B \times M_{out} \times H$ can be precisely described as follows:

$$\mathrm{O}\left(B_i, H_j, k\right) = \max_{m=0,\ldots,\mathrm{ML}-1} \mathrm{I}\left(B_i, H_j, stride \times k + m\right), \tag{14}$$

where the $O$ is the output function, $I$ denote the input function and $ML$ denotes the max length. By using Max Pooling, we transform the sentence feature vector matrix from the token level to the sentence level. Furthermore, dropout is adopted to prevent overfitting.

#### 4) LayerNorm AND ReLU

The mean value of each dimension is calculated by normalizing the sentence feature vector matrix in the channel direction.

$$\mathrm{y} = \frac{x - E[x]}{\sqrt{Var[x] + \varepsilon}} * \gamma + \beta. \tag{15}$$

Through the excitation function of the ReLU, it applies the rectified linear unit function in an element-wise manner:

$$\mathrm{ReLU}(x) = \max(0, x). \tag{16}$$

#### 5) LINEAR MODEL

This criterion combines *nn.LogSoftmax*() and *nn.NLLLoss*() in one single class. The loss can be described as follows:

$$\begin{aligned} \mathrm{loss}(x, c) &= -\log\left(\frac{\exp(x[c])}{\sum_j \exp(x[j])}\right) \\ &= -x[c] + \log\left(\sum_j \exp(x[j])\right). \end{aligned} \tag{17}$$

In the case of the weight argument being specified, it is as follows:

$$\mathrm{loss}(x, c) = weight[c]\left(-x[c] + \log\left(\sum_j \exp(x[j])\right)\right). \tag{18}$$

## IV. SIMULATION EXPERIMENTS AND RESULTS ANALYSIS

### A. DATASET

The Chinese SanWen dataset contains nine relationships between 837 Chinese literature articles, 695 of which are for training, 84 are for testing and the remaining 58 are for validation [27]. It is a discourse-level dataset from hundreds of Chinese literature articles. By using a heuristic tagging method and a machine auxiliary tagging method to solve the problem of data inconsistency, SanWen is a high-quality dataset.

### B. DATA PREPROCESSING

To facilitate the training model, we need to follow the following structure: input IDs, input sentence number, attention mask, token type IDs, label relational markers, ent1, and ent2 to store the sentence features in the example format.

Based on the extracted features, the required datasets are further processed, and the structure of all input IDs, all attention masks, all token type IDs, all labels, all entity seg pos, all entity span1 pos, and all entity pans2 pos are finally used to store the original data packets in dataset format.

Data preprocessing includes the following three stages: sentence, relational label, and entity pair extraction; entity pair initial and terminal position extraction; and entity pair position sequence data representation.

### C. EXPERIMENTAL ENVIRONMENT

The experimental environment is set as follows: ubuntu14.04os/windows10, pytorch0.4.1, python 3.6, an Nvidia 1080ti graphics card and 32g of memory. 1. To prevent the occurrence of an overfitting phenomenon, dropout is utilized during the training process with a probability of 0.8. 2. For the optimization method, the initial learning rate is set to 2e-5 with AdamW optimization. 3. Meanwhile, we use ReLU as the activation function. 4. In addition, the batch size is 8(batch size) and the program is trained for 20 epochs. 5. The vector dimension after the Embedding layer is 768 dimensions.

### D. EVALUATION METRICS

To evaluate the experimental results, we used four evaluation criteria: accuracy, recall, F1, and acc. Because relation extraction is a classification task, we need to calculate the above indicators for each category separately.

Precision equals the number of relational instances of a class that are correctly classified, divided by the total number of relational instances judged to be a class:

$$P_c = \frac{TP_c}{TP_c + FP_c}. \tag{19}$$

Recall equals the number of correctly classified relational instances of a class divided by the total number of relational instances of a class in the test set:

$$R_c = \frac{TP_c}{TP_c + FN_c}. \tag{20}$$

The F1 value is the harmonic average of the precision and recall:

$$F1_c = \frac{2P_cR_c}{P_c + R_c}. \tag{21}$$

Accuracy refers to the ratio of the correctly predicted sample size to the total predicted sample size, regardless of whether the predicted sample is positive or negative:

$$acc = \frac{\sum_{c=1}^{C} TP_c}{N}. \tag{22}$$

### E. EXPERIMENTAL RESULT ANALYSIS

In this subsection, we conduct five experiments to evaluate our proposed model. Specifically, the first experiment compared the F1 and the Area Under the Curve(AUC) values of the current best Chinese relation extraction model, the Multi-Grained Lattice(MG-lattice), to our model on the same dataset [28]. The second experiment mainly compares the loss convergence speed of the two models. The third experiment contrasts this model and analyses the effect on the value of F1 and AUC when different layers are used. The fourth experiment is to finds the best learning rate of our model. The purpose of the fifth experiment is to evaluate the classification performance of the two models using the Precision-Recall plot(P-R plot). The sixth experiment uses the statistical Precision of the at top N predictions (P@N) measure to assess the statistical significance of the proposed method. The seventh experiment mainly compares the best accuracy speed of the four models. The experiments shows that our Bert-CRE achieves state-of-the-art results on the F1 value, AUC value, P-R plot and P@N.

**TABLE 2. F1 and AUC value on different models.**

| Models | auc | F1 |
|---|---|---|
| Multi-Grained Lattice | 57.33 | 65.61 |
| Bert-CRE | **64.72** | **73.35** |

For the same trained dataset SanWen, Table 2 shows that the F1 value for the current best Chinese relation extraction model Multi-Grained Lattice is 65.61 and the AUC value is 57.33. Our model's F1 values reached 73.35 and the AUC reached 64.72. Reguarding the model's methods, BERT draws lessons from ELMO, GPT, and the Continuous Bag-of-Words Model (CBOW); and puts forward the Masked Language Model and Next Sentence Prediction [29]. How to introduce prior linguistic knowledge has always been one of the main goals of NLP, especially in deep learning, but there has been no good solution. Thus it can be seen that BERT's preprocessing model has a great influence on the F1 and AUC.

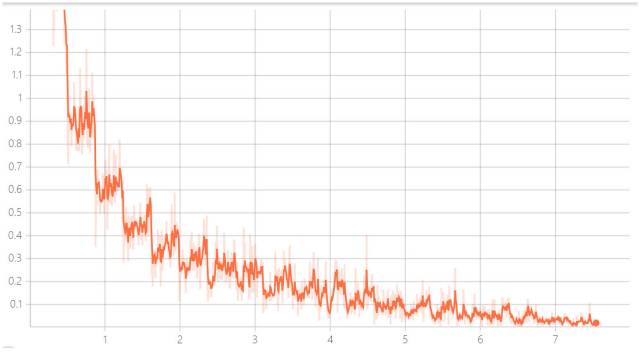



**FIGURE 5. Loss convergence comparison.**

**TABLE 3. Effects of different layers on model.**

| Model | pure Bert +sum +dropout | -sum -dropout +MaxPooler +dropout | +layerNorm +ReLU |
|---|---|---|---|
| F1 | 73.35 | **73.9** | 72.4 |
| Auc | 64.72 | 66.7 | **67.9** |

Figure 5 shows the convergence of our Bert-CRE model's loss. It can be seen that our model converges faster and the effect is better than that of the other models. From Table 3, it can be seen that the treatment of the Max Pooling layer,

layerNorm layer, and ReLU layer has a great influence on the model.

The Max Pooling layer not only calculates the maximum value in the pooled area during forwarding propagation, but it also records the position of the maximum value of the input data. It can effectively reduce redundant information and make the experimental results better.

A change in the output of one layer produces a high correlation change in the input of the next layer, especially when ReLU is used, and its output changes greatly. Then, we can reduce the effect of the covariate shift by fixing the input mean and variance of a layer of neurons.

**TABLE 4. Effects of different learning rate on model.**

| Learning Rate | 1e-5 | 2e-5 | 3e-5 | 4e-5 | 5e-5 |
|---|---|---|---|---|---|
| F1 | 72.1 | 72.6 | **73.35** | 72.0 | 70.0 |
| Auc | 62.9 | 63.6 | **64.72** | 62.4 | 61.4 |

The learning rate, as an important parameter in supervised learning and deep learning, determines whether or not the objective function can converge to the global minimum and when to the minimum. Table 4 shows that the appropriate learning rate enables the objective function to converge to a global minimum in the appropriate time. It can be seen that the different values of the learning rate have a great influence on the model. The following are the results of the pure BERT model with a fully connected layer and sum. The best learning rate is 3e-5.

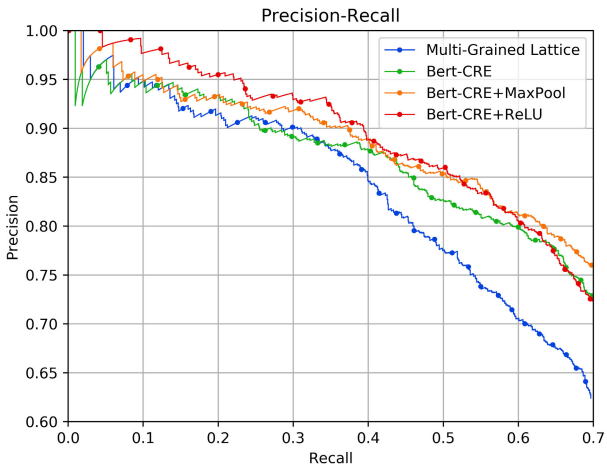



**FIGURE 6. Precision-recall.**

As figure 6 shows, our model as a whole performs better on the x and y-axes than the current best MG-lattice model. There are two problems for the current development of NLP. One is the need for a stronger feature extractor, and it is found that the Transformer is significantly stronger than the Long short-term memory (LSTM) of the MG-lattice model. The other problem is that the introduction of linguistic knowledge contained in a large number of unsupervised data is important. A considerable amount of work has attempted grafting or introducing various linguistic knowledge, but many methods are not effective. Training using this method is still very effective and very concise. It can be seen that the preprocessing model of BERT has a great influence on the precision and the recall.

We can see from the Table 5 that our proposed model is obviously superior to the current best model MG-lattice. Adding more processing layers, such as the Max Pooling layer and ReLU layer, makes the Precision of the top N predictions effect of the model is better. It can be seen that the pre-processing model is obviously improved, and the Max Pooling layer and ReLU layer improve the effect of the model.

**TABLE 5. Precision@N of models.**

| Model | P@100 | P@200 | P@300 | Mean |
|---|---|---|---|---|
| MG-Lattice | 0.941 | 0.925 | 0.904 | 0.923 |
| pure Bert<br>+sum<br>+dropout | 0.950 | 0.94 | 0.927 | 0.939 |
| -sum<br>-dropout<br>+MaxPooler<br>+dropout | 0.950 | 0.935 | 0.924 | 0.936 |
| +layerNorm<br>+ReLU | **0.990** | **0.965** | **0.950** | **0.968** |

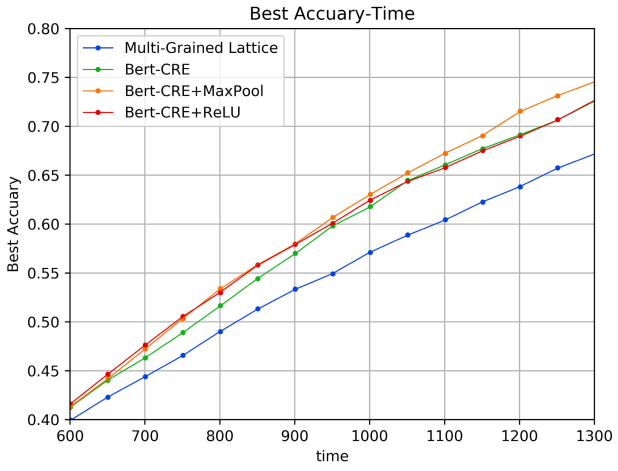



**FIGURE 7. Best accuracy-time.**

As figure 7 shows, our model as a whole performs better on the accuracy speed than the current best MG-lattice model. It can be seen that the preprocessing model of BERT has a great influence on the accuracy.

It can be seen from the results of the experimental figures and tables shows the statistical significance of our proposed method. Our model's F1 and AUC reached over the current best Chinese relation extraction model and our model converges faster and the effect is better than that of the other models. The treatment of the Max Pooling layer, layerNorm layer, and ReLU layer has a great influence on the model.

Our model as a whole performs better on the x and y-axes than the current best MG-lattice model.

Word2Vec is the pre-training model used in the MG-lattice model. BERT is the pre-training model used in our proposed model. Reguarding the training method, BERT uses the Denoising pattern to predict the words in the random MASK position. When BERT predicts the MASK dropped words, the model is trained directly through the softmax, and then it uses cross-entropy as a loss function. Word2Vec generally uses a sliding window to predict the middle word of the window or it uses the middle word to predict the words on both sides of the window. Furthermore, because of the use of words as the basic unit, the number of words is relatively large, and thus Word2Vec generally uses the stratified softmax or negative samples to conduct training. Word2Vec is also a pre-training model from the user's point of view, and the parameters of the model are the word vectors themselves. BERT as a more powerful pre-training model, and words in sentences can be output as word vectors with rich contextual information through the Transformer. Furthermore, sentence vectors can be obtained, and the pre-training of BERT is more adequate. Both BERT and Word2Vec can be part of other models, although BERT is stronger. Their biggest difference is that all Word2Vec get is a parameter matrix of the network, and the representation of each word does not change because of the different sentences in which they located. The word vector of each position in BERT has to go through a multi-layer Transformer network structure. The Transformer network structure will change the word vector of each position and the word vector of other position using the self-attention matrix, and finally the word vector of each position will fuse the information of the word vector of each position. Therefore, BERT is better than Word2Vec. The word vector of BERT in each position after many Transformer outputs, is contextual information, and it can more directly model the more distant words and word dependence, which is better than Word2Vec.

Considering the effect of the context of the Chinese relational extraction data set on the relationships among entity pairs, it should be encoded from the sentence level, and a model can not simply look up the same token with only one representation. BERT can capture contextual information more efficiently and without the limitation of distance. It has the ability to be transferred, which makes the model no longer have a limited in effect. In this paper, we introduce the BERT pre-training model to generate sentence features, and use the Transformer structure to bring the Word2Vec, which has already experienced a bottleneck, in a new direction. We can further increase the injection of position information and let the classification and judgment of relationship occur on the basis of more semantic information judgment.

## V. CONCLUSION, SUMMARY AND FUTURE WORKS

In this article, we propose a neural relation extraction model, named Bert-CRE for the relation extraction task. This paper applies the BERT pre-training model to the Chinese relation extraction, which improves the extraction accuracy and efficiency, and provides new ideas for the analysis of public security data. BERT is the latest state-of-the-art model as of October 2018, and it can solve 11 NLP tasks using pre-training and fine-tuning. Using a Transformer, BERT is more efficient and able to capture longer-distance dependencies relative to an RNN. Compared with the previous pre-training model, it captures real bidirectional context information. The operational effect is further improved by the ReLU, layer-Norm, and Max pooling. The experimental results show that our model improves the F1 and AUC of the entity relations in the extracted dataset. In the Precision-Recall plot, our model as a whole performs better on the x and y-axes than the current best MG-lattice model. In addition, the convergence effect of our model loss has achieved satisfactory performance, with faster convergence and more robustness. The Precision at the top N predictions (P@N) measure shows the statistical significance of our proposed method. In the future, we will continue to explore the application of deep learning in relational extraction, and further develop the model for public security field and Chinese structural analysis.

## ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their helpful feedback and suggestions.

## REFERENCES

[1] G. LaFree, "The global terrorism database (GTD) accomplishments and challenges," *Perspect. Terrorism*, vol. 4, no. 1, pp. 24–46, 2010.
[2] M. Khari, A. K. Garg, A. H. Gandomi, R. Gupta, R. Patan, and B. Balusamy, "Securing data in Internet of Things (IoT) using cryptography and steganography techniques," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 50, no. 1, pp. 73–80, Jan. 2020.
[3] G. G. Chowdhury, "Natural language processing," *Annu. Rev. Inf. Sci. Technol.*, vol. 37, no. 1, pp. 51–89, 2003.
[4] D. Vrandečić, "Wikidata: A new platform for collaborative data collection," in *Proc. 21st Int. Conf. World Wide Web*, 2012, pp. 1063–1064.
[5] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: A core of semantic knowledge," in *Proc. 16th Int. Conf. World Wide Web*, 2007, pp. 697–706.
[6] J. Wang, C. Jiang, K. Zhang, X. Hou, Y. Ren, and Y. Qian, "Distributed Q-learning aided heterogeneous network association for energy-efficient IIoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 4, pp. 2756–2764, Apr. 2020, doi: 10.1109/TII.2019.2954334.
[7] D. K. Gardner, C. A. Staley, and M. A. Wormley, "Question and answer system using computer networks," U.S. Patent 6 064 978, May 16, 2000.
[8] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph and text jointly embedding," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1591–1601.
[9] R. C. Schank, "Conceptual dependency: A theory of natural language understanding," *Cognit. Psychol.*, vol. 3, no. 4, pp. 552–631, Oct. 1972.
[10] R. Wu, Y. Yao, X. Han, R. Xie, Z. Liu, F. Lin, L. Lin, and M. Sun, "Open relation extraction: Relational knowledge transfer from supervised data to unsupervised data," in *Proc. Conf. Empirical Methods Natural Lang. Process., 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 219–228.
[11] X. Han, T. Gao, Y. Yao, D. Ye, Z. Liu, and M. Sun, "OpenNRE: An open and extensible toolkit for neural relation extraction," 2019, *arXiv:1909.13078*. [Online]. Available: http://arxiv.org/abs/1909.13078
[12] J. Wang, C. Jiang, H. Zhang, Y. Ren, K.-C. Chen, and L. Hanzo, "Thirty years of machine learning: The road to Pareto-optimal wireless networks," *IEEE Commun. Surveys Tuts.*, early access, Jan. 13, 2020, doi: 10.1109/COMST.2020.2965856.
[13] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. Workshop ICLR*, Jan. 2013.

[14] C. Liu, W. Sun, W. Chao, and W. Che, "Convolution neural network for relation extraction," in *Proc. Int. Conf. Adv. Data Mining Appl.* Berlin, Germany: Springer, 2013, pp. 231–242.
[15] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, "Relation classification via convolutional deep neural network," in *Proc. 25th Int. Conf. Comput. Linguistics, Tech. Papers*, 2014, pp. 2335–2344.
[16] C. N. dos Santos, B. Xiang, and B. Zhou, "Classifying relations by ranking with convolutional neural networks," 2015, *arXiv:1504.06580*. [Online]. Available: http://arxiv.org/abs/1504.06580
[17] D. Zhang and D. Wang, "Relation classification via recurrent neural network," 2015, *arXiv:1508.01006*. [Online]. Available: http://arxiv.org/abs/1508.01006
[18] P. Zhou, W. Shi, J. Tian, Z. Qi, B. Li, H. Hao, and B. Xu, "Attention-based bidirectional long short-term memory networks for relation classification," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 207–212.
[19] L. Wang, Z. Cao, G. de Melo, and Z. Liu, "Relation classification via multi-level attention CNNs," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 1298–1307.
[20] F. Ren, D. Zhou, Z. Liu, Y. Li, R. Zhao, Y. Liu, and X. Liang, "Neural relation classification with text descriptions," in *Proc. 27th Int. Conf. Comput. Linguistics*, 2018, pp. 1167–1177.
[21] F. Christopoulou, M. Miwa, and S. Ananiadou, "A walk-based model on entity graphs for relation extraction," 2019, *arXiv:1902.07023*. [Online]. Available: http://arxiv.org/abs/1902.07023
[22] C. Alt, M. Hübner, and L. Hennig, "Improving relation extraction by pre-trained language representations," 2019, *arXiv:1906.03088*. [Online]. Available: http://arxiv.org/abs/1906.03088
[23] M. E. Epstein, "Method and apparatus for embedding grammars in a natural language understanding (NLU) statistical parser," U.S. Patent 6 983 239, Jan. 3, 2006.
[24] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*. [Online]. Available: http://arxiv.org/abs/1810.04805
[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[26] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. (2018). *Improving Language Understanding By Generative Pre-Training*. [Online]. Available: https://s3-us-west-2.amazonaws.com/openai-assets/researchcovers/languageunsupervised/languageUnderstand.paper.pdf
[27] J. Xu, J. Wen, X. Sun, and Q. Su, "A discourse-level named entity recognition and relation extraction dataset for Chinese literature text," 2019, *arXiv:1711.07010*. [Online]. Available: https://arxiv.org/abs/1711.07010
[28] Z. Li, N. Ding, Z. Liu, H. Zheng, and Y. Shen, "Chinese relation extraction with multi-grained information and external linguistic knowledge," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 4377–4386.
[29] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," 2018, *arXiv:1802.05365*. [Online]. Available: http://arxiv.org/abs/1802.05365

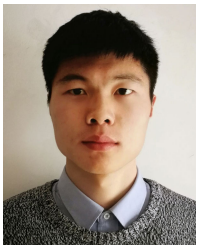
**JIAQI HOU** received the bachelor's degree in cyber security and law enforcement from the People's Public Security University of China, where she is currently pursuing the master's degree in cyberspace security and law enforcement technology. She has published some academic articles and participated in some projects. Her research interests include relation extraction, natural language processing, and knowledge graph.

**XIN LI** received the Ph.D. degree from the Department of Computer Science, Zhejiang University, in 2007. He is currently an Associate Professor at the College of Information Technology and Cyber Security, People's Public Security University of China. He has been engaged in the research on cyber security, big data, and artificial intelligence. He has published more than 30 papers in prestigious peer-reviewed journals and conferences.

**HAIPENG YAO** (Senior Member, IEEE) received the Ph.D. degree from the Department of Telecommunication Engineering, University of Beijing University of Posts and Telecommunications, in 2011. He is currently an Associate Professor at the Beijing University of Posts and Telecommunications. He has been engaged in the research on future internet architecture, network AI, big data, cognitive radio networks, and optimization of protocols and architectures for broadband wireless networks. He has published more than 90 papers in prestigious peer-reviewed journals and conferences.

**HAICHUN SUN** received the Ph.D. degree in computer software and theory from Tongji University, Shanghai, China, in 2015. She is currently an Assistant Professor at the College of Information Technology and Network Security, People's Public Security University of China, Beijing, China. She has published over ten papers on journals and conferences, such as the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS and WISE 2014. Her current research interests include information service, Petri nets, and service-oriented computing. She is a member of the Professional Committee of Internet Information Service of Chinese Association of Automation.

**TIANLE MAI** (Graduate Student Member, IEEE) is currently pursuing the Ph.D. degree with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications. His research interests include resource allocation and network association, artificial intelligence, and the future internet architecture.



**RONGCHEN ZHU** received the bachelor's degree in cyber security and law enforcement from the People's Public Security University of China. He is currently pursuing the master's degree in cyberspace security and law enforcement technology with the People's Public Security University of China. He has published some academic articles and participated in some projects. His research interests include risk analysis and assessment, Bayesian network methods and applications, and knowledge graph.