Long Text Classification Based on BERT

Ding Weijie[1,2,3*], Li Yunyi[4], Zhang Jing[5]，Shen Xuchen[6]
1. Big-data and Network Security Research Institute，Zhejiang Police College，Hangzhou 310053
2. College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023
3. Key Laboratory of Public Security Information Application Based on Big-data Architecture, Ministry of Public Security,Hangzhou 310053
4. Department of Investigation，Zhejiang Police College，Hangzhou 310053
5. Department of computer and information security,Zhejiang Police College，Hangzhou 310053
6. Science and technology management section, Xiaoshan District branch of Hangzhou Public Security Bureau, Hangzhou 311203
Corresponding Author: Ding Weijie Email:Dingweijie@zjjcxy.cn

2021 IEEE 5th Information Technology,Networking,Electronic and Automation Control Conference (ITNEC) | 978-1-6654-1599-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/ITNEC52019.2021.9587007

***Abstract*—Existing text classification algorithms generally have limitations in terms of text length and yield poor classification results for long texts. To address this problem, we propose a BERT-based long text classification method. First, we slice the long text and use BERT to encode the sliced clauses to obtain the local semantic information. Second, we use BiLSTM to fuse the local semantic information and adopt the attention mechanism to increase the weight of important clauses in the long text, so as to obtain the global semantic information. Finally, the global semantic information is input to the softmax layer for classification. Experimental results show that the proposed method achieves higher accuracy than commonly used models.**

Keywords—long text classification, BERT, BiLSTM

I. INTRODUCTION

Text classification is an important branch of natural language processing (NLP). It aims to learn and analyze classification rules using model algorithms for generalization and then applies the rules to unclassified datasets to achieve automatic classification of massive data.

Several machine learning and natural language processing algorithms have been developed thus far, and many algorithmic models have been proposed for text classification. However, the limited memorizing and forgetting capability of neural networks, as well as the weak dependency between the front and back portions of long texts, adversely affects the performance of existing text processing methods in the classification of long texts. At present, the mainstream research mainly adopts two methods to address the low efficiency of long text classification algorithms, namely data cropping to a fixed length and data slicing and reorganization[1]. As data cropping can disrupt the logical structure of the text and lead to information loss, this paper proposes a BERT-based long text classification method inspired by the idea of data slice reorganization.

Specifically, our BERT-based long text classification method uses the idea of slicing long texts into short texts[2]. Subsequently, we use BiLSTM to fuse the local semantic information and adopt an attention mechanism to increase the weight of important clauses in the long text, so as to obtain the global semantic information. Then, the global semantic information is input to the softmax layer for classification. Finally, the effectiveness of the proposed method is verified experimentally.

Although the last decade has witnessed significant advancements in terms of economic development and living standards, criminal activities, especially telecommunication crimes involving the Internet, have been on the rise[3]. The police case text records the key information of the case, and through deep excavation of the case text, the hidden clues in the case information can be analyzed, which will provide an important basis for the public security department to prevent and control crime[4]. And because the police case text needs to record complete case information, the text content is usually long, which makes automatic classification difficult.

At present, police cases are still classified manually. On the one hand, this approach requires considerable human and material resources; on the other hand, because of its strong subjectivity, this method can easily cause confusion in text categorization. Consequently, the public security department cannot grasp the actual situation of each type of case and is thus unable to formulate realistic measures to prevent and control crime. Therefore, accurate classification of long texts is the most basic and urgent requirement for public security departments. The long text classification model proposed in this paper can effectively classify case texts related to public security and automatically classify long texts of police cases.

II. RELATED WORK

Early text classification approaches were mainly based on traditional machine learning methods. Shallow learning models for text classification are faster and have significantly higher classification efficiency than manual classification methods. As one of the core problems of text classification, classifier selection and training is usually based on the frequency of words or bag-of-words features in traditional machine learning methods, and training is performed with models. Representative traditional machine learning classification methods include the support vector machine (SVM), plain Bayes algorithm, and random forest.

In 2003, Yoshua Bengio[5] proposed a neural probabilistic language model, in which the word vector representation of text considers the correlation between words, effectively solving the problem of each feature item being independent of others; this correlation is ignored in the traditional vector space model.

The unidirectional long short-term memory (LSTM) network is designed to address the problem of short memory periods of recurrent neural networks (RNNs), which makes it difficult to transfer information to farther layers. It integrates the forgetting gate, input gate, and output gate control units to realize effective sequence feature extraction. To overcome the inadequacy of traditional methods in capturing contextual information, Google launched Word2vec in 2013 and constructed two model structures, namely CBOW and Skip-gram, based on the prediction method, using the current location context for prediction.

To address the shortcomings of LSTM, which can only extract text features in one direction, the bidirectional LSTM (BiLSTM) model combines forward and backward LSTM to achieve the extraction of global contextual semantic features. Lai proposed a model based on RNNs and convolutional neural networks (CNNs) to further improve the capture and application of contextual information and word order by using bidirectional networks to expand the scope of word order retention during learning.

The most advanced model is the bidirectional encoder representation from transformers (BERT) model of Google's bi-directional transformers encoder structure, released in 2018, which improved the optimal performance of 11 NLP tasks with a pre-trained model.[6]

In 2020, Ning Jin and Chunjiang Zhao[7] proposed a short text classification method based on BiGRU_MuICNN for agricultural questions and answers. Lin Du and Dong Cao[8] proposed a short text classification method based on the BERT+BiLSTM+Attention fusion mechanism. Chenfeng Ma[9] applied deep learning to the classification of news text. Jianhao Wang and Yong Su[10] used the K-means algorithm to classify police text by selecting sample points in dense regions as clustering centers. Wenyan Wei and Xin Lv[4] jointly used an SVM classifier and a rule classifier to classify police text. Chunhui Cheng and Qinming He[11] employed a multivariate Bayesian model by adding the ratio of the current category sample size to the maximum sample size, and applied it to a dataset with category imbalance. Haoquan Li and Mengfan Shi[12] used a CNN for text classification.

III. INTRODUCTION OF THE MODEL

A. Overview of the Model

The proposed BERT-based police long text classification method consists of five main parts: text slicing, BERT pre-training coding, BiLSTM semantic fusion, attention mechanism weight calculation, and softmax classification. Fig.1 shows the overall model structure.

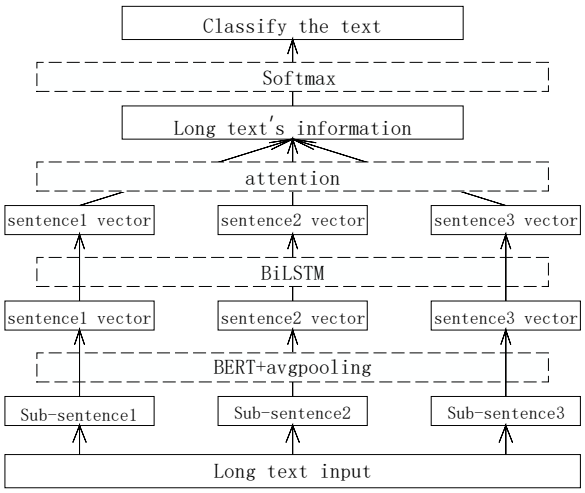

Fig. 1. Flow chart of model structure bilateral

The model first slices the long text of the police case into multiple clauses and uses the BERT language model to encode each clause in order to obtain the local semantic information of the long text. Then, it uses BiLSTM to fuse the global semantic information between multiple clauses and adopts the attention mechanism to assign corresponding weights to different clauses. Finally, it uses the softmax layer to classify the text.

B. Long Text Slicing

We use both fixed-length and quantitative cuts to slice long texts. For fixed-length cuts, all long texts are set to yield the same length of clauses, and different long texts may yield different numbers of clauses after the cut. For quantitative cuts, all long texts are set to yield the same number of clauses, and different long texts may yield different lengths of clauses after the cut. The results of using fixed-length and quantitative cuts for two different texts are shown in Fig.2 and Fig.3, respectively.

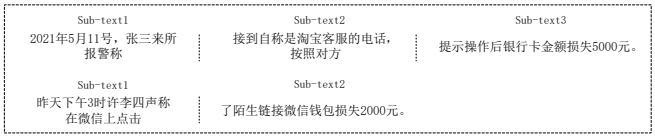

Fig. 2. Fixed-length cut

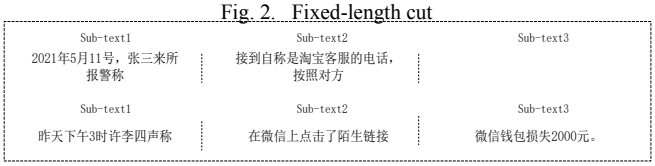

Fig. 3. Quantitative cut

C. BERT Encoding to Obtain Local Semantic Information

The BERT pre-trained language model consists of a bi-directional transformer encoding layer structure, which employs post-text semantics to improve the pre-trained representation compared to the one-way transformer encoding approach of the OpenAI GPT model. It pre-trains a deep bi-directional representation by jointly adjusting the context in all layers. Further, it uses a masked language model (MLM) to replace 15% of the words in the input text with random words.This model forces the transformer to consider the influence of contextual information when encoding words by continuously adjusting the parameters to make the model predict the [mask] lexical items with optimal accuracy.

The Chinese BERT model treats each Chinese word as a token, and for a given t-th clause containing N Chinese words, $sen_t=w_1,w_2,...w_N$, where w_i ($1\le i\le N$) represents the i-th word in the input text, the Chinese BERT model splits the input vector of w_i into three parts: semantic vector, position vector, and paragraph vector. Then, it performs three embeddings as the input of the model:

$$E(w_i)=E_{token}+E_{position}+E_{segmen} \quad (1)$$

The query vector q, key vector k, and value vector v of w_i are obtained by the dot product of $E(w_i)$ with different weight matrices, as follows:

$$q = E(w_i)\cdot W_Q \quad (2)$$

$$k = E(w_i)\cdot W_K \quad (3)$$

$$v = E(w_i)\cdot W_V \quad (4)$$

On this basis, the transformer encoding structure uses a self-attention mechanism to calculate the correlation degree among the text tokens, and sets the weight of the tokens in text semantic analysis based on the correlation degree, so that each token can be dynamically reflected in the final word vector of the text. Under the condition that the input vector dimension d_k is introduced and $\sqrt{d_k}$ is used as the penalty factor, the self-attention value between w_i and each word in the input clause sen_t is obtained by computing the dot product between the

query vector q and the key vector k. Further, the semantic information of w_i in the current context is obtained by weighting and summing the self-attention value with the value vector v of each word, as follows:

$$E_{attention}(w_i) = \sum_{i=1}^{N} softmax(\frac{q_i \cdot k_i}{\sqrt{d_k}})v_i \quad (5)$$

Finally, a feedforward neural network is used to enrich the semantic information of w_i, as follows:

$$E_{FFN} = W_i \cdot E_{attention}(w_i) + b_i \quad (6)$$

The above-mentioned calculations are carried out in several sub-layers of BERT to finally obtain the coded information of BERT:.

$$Output\_bert = (E(cls), E(w_1), \dots E(w_N)) \quad (7)$$

where E(cls) represents the semantic vector of input clause sen_t and $E(w_i)(1 \le i \le N)$ represents the semantic vector of each word in sen_t. Current studies on text classification directly use E(cls) as the input vector of a downstream task. To achieve data dimensionality reduction while fully exploiting the semantic information of each word in the input clause, this study draws on the idea of the average pooling layer in CNNs [13] and performs the average pooling operation on ($E(w_1)$, $E(w_2)$, ...$E(w_N)$), as follows:

$$avg = \frac{\sum_{i=1}^{N} E(w_i)}{N} \quad (8)$$

In addition, to avoid losing the input clause semantic vectors obtained by the BERT model during the training process, the average pooled word semantic vector, avg, is fused with the sentence semantic vector, E(cls), as the final semantic information of the input clause, i.e., the local semantic information of the long text, as follows:

$$local\_info_t = avg + E(cls) \quad (9)$$

D. BiLSTM Fusion Semantic Information

The BiLSTM model consists of forward and backward LSTMs. Through the gate control unit design, the LSTM iteratively updates the information memorized in the neural network by forgetting some of the information in the neurons. Thus, it can realize the memory function for longer texts[14]. Compared with the one-way LSTM model, which can only encode the text information from front to back, the BiLSTM model utilizes global semantics via bi-directional encoding followed by splicing.[15] The flowchart of the BiLSTM model for fusing local semantic information is shown in Fig.4.

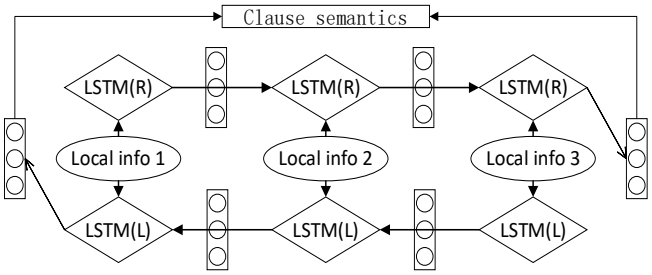

Fig. 4. BiLSTM fuses pre- and post-textual semantic information

When encoding the semantic information of the t-th clause, the BiLSTM model encodes the forward and backward order among the clauses, respectively, and splices the forward and backward encoding vectors to obtain the semantic information of the t-th clause, at which time the t-th clause has fused the semantic information of the other clauses, as follows:

$$\overrightarrow{h_t} = \overrightarrow{LSTM}(local\_info_t) \quad (10)$$

$$\overleftarrow{h_t} = \overleftarrow{LSTM}(local\_info_t) \quad (11)$$

$$h_t = [\overrightarrow{h_t}, \overleftarrow{h_t}] \quad (12)$$

E. Attentional Mechanisms

After the BiLSTM fusion of the semantic features of each clause, each clause is fused with the semantic information of the other clauses, and the semantic information of the original long text can be obtained by combining the semantic information of each clause. However, as the importance of each clause to the original long text is different, simple summation is not possible. To enhance the influence of the important clauses on the semantics of the original long text and weaken the influences of clauses with weak global semantic relevance, this study adopts an attention mechanism[16] to assign different weights to different clauses. The weights of T clauses are calculated as follows:

$$\alpha_t = \frac{exp\ (h_t)}{\sum_{t=1}^{T} exp\ (h_t)} \quad (13)$$

The semantic information of each clause is weighted and summed to obtain the semantic information of the original long text, as follows:

$$global\_info = \sum_{t=1}^{T} \alpha_t \cdot h_t \quad (14)$$

F. Softmax layer

After the global semantic fusion of the long text is completed, the semantic information of the original long text is downscaled using the feedforward neural network, and the probability of the original long text belonging to each category is calculated using the softmax function[17], as follows:

$$FFN\_info = ReLU(W_{FFN} \cdot global\_info + b_{FFN}) \quad (15)$$

$$P = \frac{exp\ (FFN\_info_i)}{\sum_{i=1}^{FFN\_info\_dim} exp\ (FFN\_info_i)} \quad (16)$$

To train the updated model parameters, this study defines the minimization loss function as follows:

$$Loss(Y, P) = -\frac{1}{S}\sum_{i=1}^{S} 0 \sum_{j=1}^{C} y_{ij} log\ (P_{ij}) \quad (17)$$

where S represents the number of samples in the batch training sample set, C is the number of categories, y_{ij} is the true probability that the i-th sample belongs to the j-th category, and P_{ij} represents the probability that the model will predict the i-th sample to belong to the j-th category.

IV. Application of Model to Police Text Data

A. Police Text Data Features

The police case text contains a detailed description of the crime. Hence, the text content is generally long. According to statistics, more than 50% of the police text length exceeds the optimal range of the BERT model. The neural network for obtaining information shows poor performance. Information loss can easily occur and reduce the learning accuracy. The proportions of the data volume of different length intervals obtained by the research statistics are listed in Table I.

B. Experimental Data

To verify the effectiveness of the BERT-based long text classification method for police cases in actual public security scenarios, this study selects two types of text data of police cases involving online and non-internet cases actually entered by public security departments as experimental data. After text pre-processing, a total of 500,000 valid data are obtained for subsequent classification tasks. In this study, the experimental data are divided into training and validation sets in the ratio of 8:2. Meanwhile, to maintain consistency of the training effect of the model on the two types of texts, this study ensures that

the positive and negative samples in the training set are in the ratio of 1:1 when dividing the data. The specific data distribution is summarized in Table II.

TABLE I. SAMPLES WITH INTERVALS OF DIFFERENT LENGTHS

Text length	Percentage
<100	8%
100-512	34%
512-1024	47%
>1024	11%

TABLE II. EXPERIMENTAL DATA

Total Sample Size	Validation Set	Train Set	
		Positive Sample	Negative Sample
500000	100000	200000	200000

C. Experimental Environment and Parameters

In the training process, the transformer encoding layer structure in the BERT model has 12 layers, the number of heads in the self-attention mechanism is 12, and the number of dimensions is 768. The Adam optimizer is used to update the model parameters and set the initial learning rate to 0.001, the learning rate decay coefficient to 0.95, the number of training rounds epoch to 6, and the dropout probability to 0.1. In addition, to improve the model stability, this paper selects 10% of the data for preheating.

TABLE III. MODEL PARAMETER SETTING

Parameters	Values
Transformer Layers	12
Attention Heads	12
Hidden Size	768
Optimizer	Adam
Initial Learning Rate	0.001
Learning Decay Rate	0.95
Epoch	6
Batch Size	8
Attention Probs Dropout Prob	0.1
Hidden Dropout Prob	0.1

D. Evaluation Indicators

In this study, the micro-average accuracy rate is selected as the evaluation index for the classification effect of police cases. It is calculated as the ratio of the number of correctly classified samples to the total number of samples as follows:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (18)$$

where TP represents the number of samples for which the model predicts positive samples as positive classes, TN represents the number of samples for which the model predicts negative samples as negative classes, FP represents the numbers of samples for which the model predicts positive samples as negative classes, and FN represents the number of samples for which the model predicts negative samples as positive classes.

E. Experimental Results and Analysis

To determine the optimal classification performance of the proposed model, we conducted several experiments on quantitative and fixed-length cuts using different parameter settings. Thus, we obtained the trend of the micro-average accuracy of the model with different numbers of clauses and different clause lengths, as shown in Fig.5 and 6.

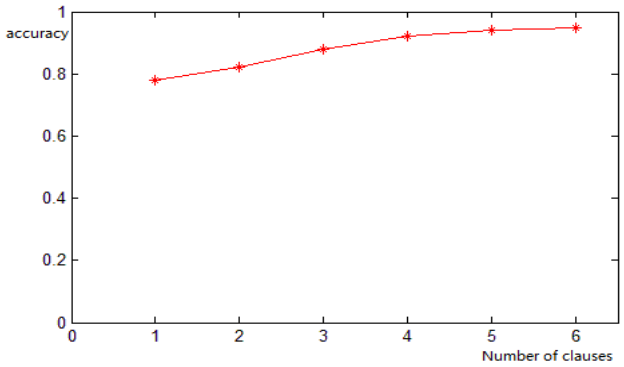

Fig. 5. Trend of accuracy of quantitative cuts

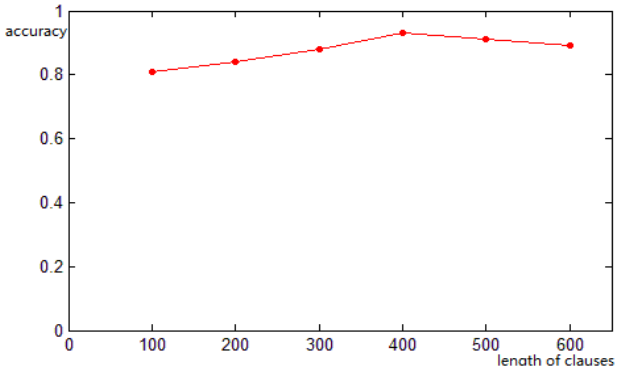

Fig. 6. Trend of accuracy of fixed-length cuts

The experimental results showed that when the long text is cut into 5 clauses with the quantitative cut, the accuracy rate is stable around 95%. By contrast, with the fixed-length cut, there is a turning point when the clause length of the long text is 400, and the classification effect is optimal.

TABLE IV. ACCURACY WITH DIFFERENT NUMBERS OF CLAUSES

算法模型	1	2	3	4	5	6
BERT+avgpooling+ BiLSTM +attention	0.78	0.82	0.88	0.92	0.94	0.95
BERT+BiLSTM	0.75	0.79	0.81	0.84	0.87	0.91
BERT	0.74	0.77	0.79	0.81	0.82	0.88
BiLSTM	0.69	0.73	0.77	0.79	0.81	0.83

In addition, we compared the effects of several commonly used text classification methods under fixed-length and quantitative cutting. The trend of accuracy of these methods with different numbers of clauses is shown in Table IV and Fig.7.

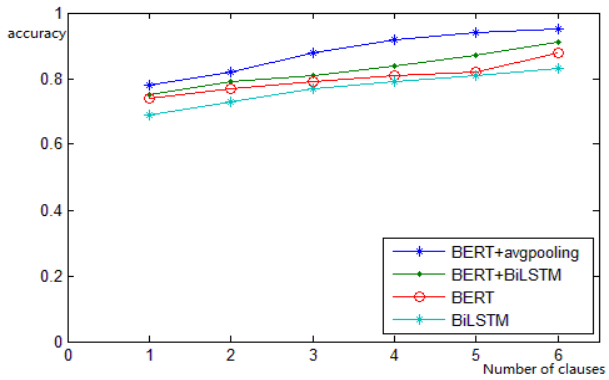

Fig. 7. Accuracy with different numbers of clauses

The experimental results showed that the proposed method outperforms several baseline methods in the actual police case text data classification task with each number of clauses, and its advantage is the most obvious when the number of clauses is 4 (8% improvement in accuracy compared to the commonly used BERT+ BiLSTM). As the number of clauses increases in the range of [1,6], the accuracy of each algorithm keeps improving. Nevertheless, the following order is always maintained: BERT+avgpooling+ BiLSTM +attention > BiLSTM +attention > BERT > BiLSTM. This is because the BERT pre-trained model is more capable of semantic representation and beneficial for lower-level classification tasks compared to BiLSTM, which selectively forgets partial clause information when fusing global semantics.

V. Conclusion

To improve the efficiency and effectiveness of long text classification in order to facilitate rapid classification of the large amount of police texts accumulated in public security departments and thus improve the utilization rate of text data by public security authorities, this paper proposed a BERT-based long text classification method. First, the long text was sliced, after which BERT and BiLSTM were employed to obtain the local and global semantic information, respectively. Then, an attention mechanism was adopted to assign corresponding weights to different clauses. Finally, the global semantic information was input to the softmax layer for category classification. The test results on the police text dataset showed that the classification efficiency of the proposed model is higher than that of the baseline method. In summary, our model has certain practical significance and can effectively improve the classification efficiency of police text.The shortcomings of the model include the fact that the text cutting method is limited to fixed length and quantitative, and no better cutting method is explored. Also, the amount of training and validation data used in the model is not large enough. We will try to optimize the long text scoring method in Follow-up studies, such as the targeted selection of scoring methods based on the preliminary overall analysis results.

VI. Acknowledgment

This research work was partly supported by Natural Science Foundation of Zhejiang Province (Grant No. LGF19G010001 and LGF21G030001) and Basic Project of Strengthening Police by Science and Technology of the Ministry of Public Security (Grant No.2020GABJC35).

References

[1] Pappagari R , Zelasko P , Jesús Villalba, et al. Hierarchical Transformers for Long Document Classification[C]// 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). IEEE, 2020.

[2] ZeLong Li. Research on fast and precise classification algorithm of long text based on FastText[D].Zhejiang University,2018.

[3] Jiazhong Zhang, Yanyan Duan. Path analysis of the construction of social security precise prevention and control system under the background of big data[J].Jingyue Journal, 2018, (1): 117-122.

[4] Wenyan Wei, Xin LV,Yan Gao. Application of Text Mining Technology in the Field of Public Security[J]. Journal of Hunan Police Academy, 2017, 29(03): 98-104.

[5] Bengio Y , Réjean Ducharme, Vincent P , et al. A Neural Probabilistic Language Model.[J]. Journal of Machine Learning Research, 2003.

[6] Devlin J , Chang M W , Lee K , et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[J]. 2018.

[7] Ning Jin, Chunjiang Zhao, et al. Classification Technology of Agricultural Questions Based on BiGRU_MulCNN[J]. Transactions of The Chinese Society of Agricultural Machinery, 2020, 051(005): 199-206.

[8] Lin Du,Dong Cao,et al.Extraction and Automatic Classification of TCM Medical Recaords Based on Attention Mechanism of Bert and Bi-LSTM[J]. Computer Science, 2020,47(S2):416-420.

[9] Chenfeng Ma,Hybrid Deep Learning Model for News Classification[D]. Shandong University,2018.

[10] Jianhao Wang,Yong Su. Application in Case Prediction Based on the K-means Algorithm[J].Computer and Digital Engineering, 2019, 47(8).

[11] Chunhui Cheng, Qinming He. Naive Bayes based criminal text classification of unbalanced classes.Computer Engineering and Applications[J]. Computer Engineering and Applications, 2009, 45(035):126-128,131.

[12] Wuquan Li,Mengfan Shi, et al. Application of Convolutional Neural Network in Case Classification[J]. Computer Engineering and Software, 2019, 040(004): 222-225.

[13] Yanying Mao. Long Text Emotion Classification Method based on the Attention Double-layer LSTM[J]. Journal of Chongqing College of Electronic Engineering, 2019, 028(002):118-125.

[14] He K , Zhang X , Ren S , et al. Deep Residual Learning for Image Recognition[J]. 2016.

[15] Ghaeini R , Hasan S A , Datla V , et al. DR-BiLSTM: Dependent Reading Bidirectional LSTM for Natural Language Inference[C]// Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). 2018.

[16] Jun-Hua G U , Wei-Tao P , Na-Na L I , et al. Sentiment classification method based on convolution attention mechanism[J]. Computer Engineering and Design, 2020.

[17] Kingma D , Ba J . Adam: A Method for Stochastic Optimization[J]. Computer Science, 2014.