Research on fusion model of BERT and CNN-BiLSTM for short text classification

Chao Yang [1], Xiaotian Wang [2] , Mengyu Li [2a*], Ji Li [2]
[1] State Grid Hebei Electric Power Co., Ltd., Shi Jiazhuang, 050000, Hebei, China
[2] State Grid Hebei Marketing Service Center, Shi JiaZhuang, 050000, Hebei, China
[a*] napoleonlmy@126.com

***Abstract*—With respect to short texts with high information content, unstructured and non-standard, a text classification model (BERT-CNN-BiLSTM) based on the fusion of BERT model and BiLSTM network with convolutional neural network is proposed. To improve data processing efficiency and classification precision, word vectors are trained in BERT and used as the embedding layer of the model. The embedding layer is utilized to retain semantic information and the semantic representation of words is enhanced. CNN is applied to extract the local semantics of text. Meanwhile, gated linear unit (GLU) is used to optimise the CNN, and gradient dispersion is reduced. BiLSTM is designed to acquire contextual information about the text. Text classification is better implemented. The experimental results show that better results are obtained by BERT training data as word vectors. The BERT-CNN-BiLSTM has significantly improved in terms of classification precision, recall and F1 than the CNN, the BERT-CNN, et al. Precision, recall and F1 values are improved by at least 1.44%, 1.66% and 1.69%, respectively.**

Keywords- Short text classification; Fusion model; BERT; CNN; BiLSTM

I. Introduction

With the rapid development of the Internet, big data, and social platforms, service levels have also become an important factor in user satisfaction with products. In electric service, a large number of customer work order texts are generated. Customer work order text data has the characteristics of multi-spoken language, a large amount of information and unstructured. Manual processing is extremely inefficient and it is clear that relying on manual classification is not desirable.

In short text classification tasks, traditional machine learning algorithms require the manual extraction of text features for text classification. However, the interrelationships and interactions between features are ignored[1]. With the rapid development of deep learning, text classification problems can be better handled. Among them, Convolutional Neural Network (CNN) is a good solution for text feature extraction for short text classification. In response to the existence of semantic relationships between words in short texts, the self-linking and interlinking mechanisms in the implicit layer of recurrent neural network (RNN) models are able to better read and memorize the information above[2]. The short text classification method of extracting feature words by word2vec and then further extracting high-level features from CNNs was proposed by Duan et al.[3]. This method effectively improves the two-level classification accuracy of Internet short texts. The CNN_BiLSTM_Attention hybrid model was proposed by Wu et al.[4]. This model fuses key pattern information and global structure information at different levels to obtain the final text representation, which improves classification accuracy. A model of feature fusion to extract features of text from different levels is proposed by Yang et al.[5], which is good for text classification.

Word embedding techniques have been evolving and gradually becoming mature. The integration of deep learning and word embedding learning has achieved great results. Recently, a text classification method based on BERT and CNN was proposed by Chen et al.[6]. Through the training of a large corpus, different semantic representations of words are considered. Finally, the output of a dynamic word vector is formed. A text classification based on BERT with BiLSTM fused attention mechanism was proposed by Du et al.[7]. The model combines the BERT model for short text vectors as text input and the classification precision is improved. An attention mechanism based on the fusion of BERT and Bi-GRU was proposed by Sun et al.[8]. The model uses a word vector of BERT optimised words, a Bi-GRU network to extract key words for text classification. A text multi-label classification method combining BERT and tag semantic attention was proposed by Lv et al.[9] to improve the classification precision. The BGCN model was proposed by Cheng et al.[10]. The BGCN model improves detection efficiency by combining a trigger word detection model based on BERT word embedding vectors with GCN.

In this paper, a fusion model based on BERT-CNN-BiLSTM is proposed to improve the effectiveness of text classification. A BERT model is used to train word vectors as input vectors to the text and a CNN is used to extract local features of the text. GLU is applied to optimize the CNN to reduce the gradient dispersion problem. The output vector of the CNN is also used as the input to the BiLSTM, which is used to obtain the contextual information of the text. A Softmax classifier is used for text classification.

II. BERT-CNN-BiLSTM fusion model

In this paper, the BERT-CNN-BiLSTM fusion model mainly consists of BERT word embedding layer, CNN layer for local feature extraction, BiLSTM layer for contextual feature extraction and classification layer. The model structure is shown in Fig. 1.

2023 4th International Conference on Computer Engineering and Application (ICCEA) | 979-8-3503-4754-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICCEA58433.2023.10135222

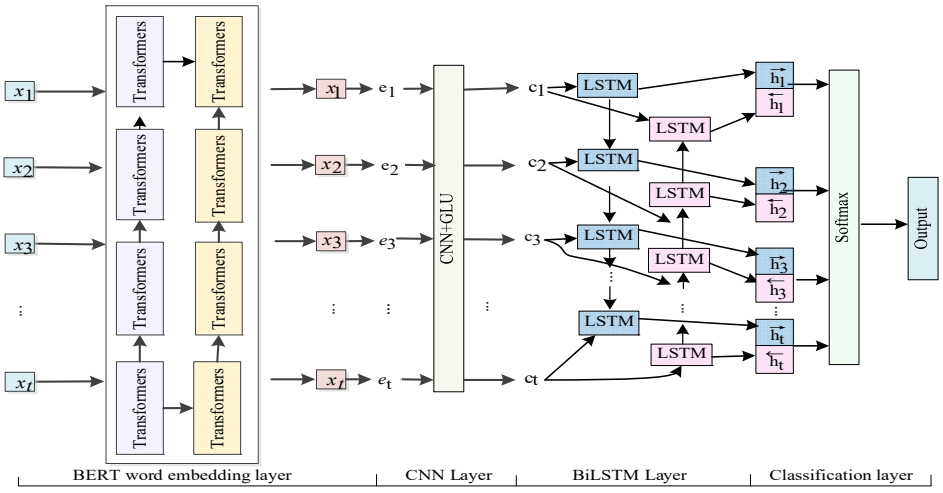

Fig.1 Structure of BERT-CNN-BiLSTM hybrid model

A. BERT word embedding

Pre-trained language models are gradually yielding good results in natural language processing tasks. BERT mainly consists of two self-supervised tasks, Masked Language Model(MLM) and Next Sentence Prediction(NSP). MLM means replacing some words with Mask words with a small probability from the input at training time. NSP is judging another sentence based on one of them. The results of the BERT model are shown in Fig. 2.

From Fig. 2, let the sentence to be entered be $X=x_1,x_2,\cdots,x_n$, x_i is the i-th word. Meanwhile, the input BERT model requires three vectors of words: word embedding vector, position embedding vector and segmentation embedding vector. The sentence output vector is $x=\{x_1,x_2,\cdots,x_n\}$, and the output is used as the input to the CNN module.

B. Local feature extraction

1) Convolutional Neural Network

CNNs were first applied in the field of computer vision and image processing. In recent years, it has been widely used in natural language processing tasks[2]. CNN is mainly used to extract features from the input data, using pooling layers to select and filter the features extracted from the convolutional layers, and multiple pooling layers are concatenated and finally output them. The CNN model is shown in Fig. 3.

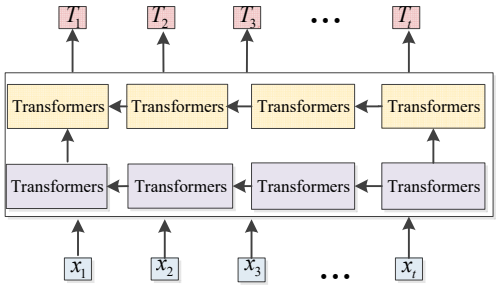

Fig. 2 the BERT model architecture

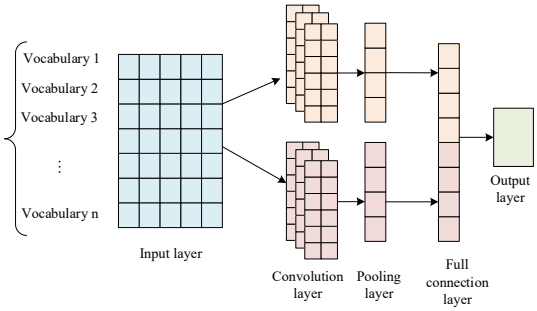

Fig. 3 the CNN model architecture

2) Gated linear unit

The introduction of threshold control into the CNN constitutes a Gated Linear Unit (GLU) and a Gated Hyperbolic Tangent Unit (GTU). As GTU is prone to the problem of abrupt gradient disappearance, GLU is used as an activation function to control the transfer of information in the hierarchy. The GLU takes the result of the convolution and passes it through linear mapping and S-shaped gating respectively, and multiplies the output of both as the input to the next layer.

$$h_l(x)=(x*W+b)\otimes \sigma(V*x+b) \tag{1}$$

where W and V are the weights, x is the input word vector matrix, and the parameter b is the bias term, σ is a Sigmoid function.

C. Contextual feature extraction

Many variants such as GRU and LSTM have evolved to overcome the problems of gradient disappearance and explosion during the backpropagation of RNN in updating the network parameters. The LSTM network gating mechanism is utilized in this paper. The exact formula is shown below.

$$i_t=\delta(W_i\cdot[h_{t-1},x_t]+b_i \tag{2}$$

$$f_t=\delta(W_f\cdot[h_{t-1},x_t]+b_f) \tag{3}$$

$$\tilde{c}_t=\tanh(W_c\cdot[h_{t-1},x_t]+b_c) \tag{4}$$

$$c_t=f_t\odot c_{t-1}+i_t\odot\tilde{c}_t \quad (5)$$

$$o_t=\delta(W_o\cdot[h_{t-1},x_t]+b_o) \quad (6)$$

$$h_t=o_t\odot \tanh(c_t) \quad (7)$$

where i_t is the input gate at moment t. f_t is the forgetting gate at moment t. c_t is the state of the memory cell at moment t. o_t is the output gate at moment t. h_{t-1} is the input of the previous cell. x_t is the input at moment t. δ is the sigmoid function. w and b are the weights and biases, respectively.

BiLSTM can fuse the information before and after in the sequence in the memory information at the same time. The difficulty that LSTM models cannot capture contextual information due to serialization processing problems is solved. Theoretically, the classification precision can be improved. The structure of the BiLSTM model is shown in Fig. 4, and the BiLSTM is calculated as follows.

$$\vec{h}_t=LSTM(\vec{h}_{t-1},u_t) \quad (8)$$

$$\overleftarrow{h}_t=LSTM(\overleftarrow{h}_{t-1},u_t) \quad (9)$$

$$h_t=w_t\vec{h}_t+v_t\overleftarrow{h}_t+b_t \quad (10)$$

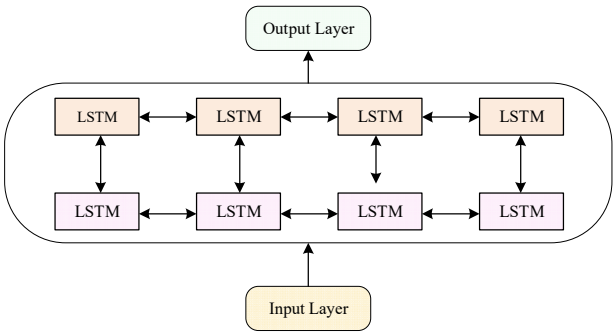

Fig. 4 the BiLSTM model structure

Where $\vec{h}_t$, $\overleftarrow{h}_t$ are the outputs obtained for the forward and backward LSTM at moment t respectively. The output states of the BiLSTM are h_t. w_t, v_t, b_t is the weight matrix and bias weights. u_t is the input to the LSTM at moment t.

D. Classifier

The softmax classification layer analyses features from a global perspective to complete the text classification task. Output the probability distribution of the category whose probability of classifying x into category j is as follows:

$$p(y^i=j|x^i;\theta)=\frac{exp(\theta_j^T x^i)}{\sum_{n=1}^{k} exp(\theta_n^T x^i)} \quad (11)$$

where θ is the parameter in training and k is the classification class.

III. Experimental results and analysis

A. Experimental environment

In this paper, the experiments are implemented on a Windows 7 operating system with an Intel(R) Core(TM) i5-5200U GPU device. Experiments are conducted using Python3.7, and the GPU version of the deep learning framework.

B. Dataset pre-processing

The information on the three datasets used in this paper is shown in Table 1. Each dataset is manually labeled with the category to which it belongs. The dataset is divided into a training set and a validation set in a ratio of 8:2.

Table 1 Statistical information on the datasets

Datasets	Title	Train set	Test set	class	total
Dataset1	service application work orders provided by an electricity company in Hebei province	8000	2000	6	10000
Dataset2	Taobao customer service	16000	4000	5	20000
Dataset3	service quality evaluation of a hotel	40000	10000	5	50000

C. Training parameter setting

In this paper, the word vectors trained by the keras-bert model are used to obtain the word vector document. The network layer is 768 dimensional. The CNN implicit layer dimension is 128. GLU is the activation function. The BiLSTM implicit layer dimension is 256. The optimizer is Adamax. The learning rate is 0.001. Dropout is 0.3. Epoch is 10. Earlystop is 3. If the loss rate does not drop for 4 consecutive times, training is stopped, the optimal epoch is recorded, and the optimal model is saved.

D. Evaluation criteria

Precision, Recall and F1-Score are used as the evaluation metrics respectively. The classification performance of the BERT-CNN-BiLSTM model is evaluated by the following formula.

$$Precision=\frac{TP}{TP+FP} \quad (12)$$

$$Recall=\frac{TP}{TP+FN} \quad (13)$$

$$F1=2\cdot\frac{Precision\cdot Recall}{Precision+Recall} \quad (14)$$

Where TP indicates a positive actual category and a positive categorical category, FP indicates a negative actual category and a positive categorical category, FN indicates a positive actual category and a negative categorical category, and TN indicates a negative actual category and a negative categorica category.

E. Comparison of different optimisation operations

In the CNN-BiLSTM model, the ReLU activation function, GLU and GTU were used to compare the effects of different activation operations on text classification performance. The

F1-Score comparison of the CNN-BiLSTM model with different activation operations on different datasets is shown in Fig.5.

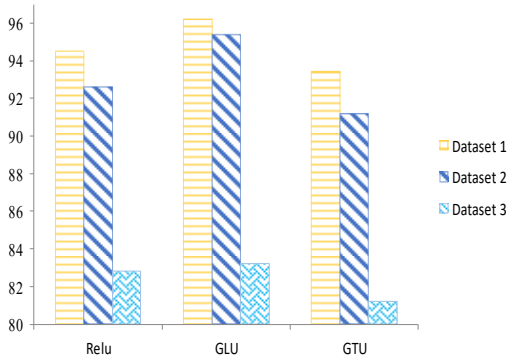

Fig. 5 Comparison for different activation layers

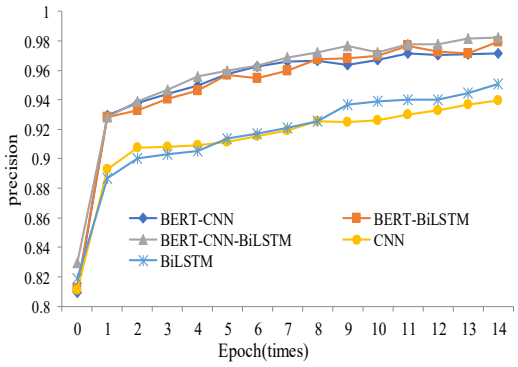

Fig. 6 Comparison of the precision rates of models

As can be seen in Fig. 5, the F1-Score values using GLU are 1.71% and 2.80% higher than the ReLU activation function, respectively. The F1-Score values are higher than the GTU values. Thus GLU is better for optimization in CNN.

F. Comparison experiments with different classification models

To exhibit the superiority of the BERT-CNN-BiLSTM, five models are selected for comparison of the precision. The check-set accuracy for the training process of dataset one is shown in Fig. 6.

From Fig. 6, it can be seen that the precision of BERT-CNN-BiLSTM under the condition that the BERT model is used as the word vector is higher and superior to the other methods.

To verify the advantage of BERT embedding layer training word vectors and the capability of BERT-CNN-BiLSTM, BERT-CNN-BiLSTM is compared with CNN, BiLSTM, BERT, CNN-LSTM, BERT-CNN and BERT-BiLSTM under different datasets. The experimental results are shown in Table 2.

In the comparison experiments, the local features of the text can be well extracted by CNN when using the BERT model to obtain word vectors as the embedding layer. However, the contextual information of the text could not be extracted. In contrast, the BiLSTM can extract the contextual information of the text well, while ignoring the local features of the text. Thus the fusion of CNN and BiLSTM can extract both the textual information and the contextual information of the text. As shown in Table 2, the BERT-CNN-BiLSTM improved over BERT-CNN by 0.37%, 4.25% and 5.01% in F1 value, while improving over BERT-BiLSTM by 1.62%, 1.68% and 2.64% in F1 value, respectively. It can be seen that BERT-CNN-BiLSTM performs better than other methods under different datasets.

Table 2 Comparison of model evaluation criterias under different datasets

Datasets	Evaluation criteria	CNN	BiLSTM	BERT	BERT-CNN	BERT-BiLSTM	CNN-LSTM	BERT-CNN-BiLSTM
Dataset1	Precision	0.9370	0.9434	0.9471	0.9584	0.9452	0.9477	0.9621
	Recall	0.9384	0.9429	0.9469	0.9589	0.9480	0.9460	0.9626
	F1	0.9370	0.9431	0.9461	0.9586	0.9461	0.9430	0.9623
Dataset2	Precision	0.8450	0.8630	0.8977	0.9126	0.9381	0.9378	0.9549
	Recall	0.8442	0.8639	0.8960	0.9130	0.9387	0.9380	0.9555
	F1	0.8444	0.8634	0.8930	0.9127	0.9384	0.9383	0.9552
Dataset3	Precision	0.7612	0.7941	0.7977	0.8828	0.8067	0.8259	0.8327
	Recall	0.7612	0.7941	0.7960	0.8830	0.8065	0.8257	0.8332
	F1	0.7612	0.7941	0.7930	0.8829	0.8066	0.8260	0.8330

IV. Conclusion

In this paper, word vectors of text are obtained by using BERT word embedding training as the input of the model for short text classification of work orders. The experimental data shows that the word vectors obtained from the BERT model have better results for work order text classification. CNN combined with BiLSTM model is better to extract local features and contextual information of text. The GLU is utilized as an activation function for CNN to better mitigate the gradient dispersion problem. Compared with other methods, the experiments show that the data processing efficiency and model classification accuracy are improved. The effectiveness and feasibility of the BERT-CNN-BiLSTM are verified.

Acknowledgments

This work was financially supported by the Science and Technology Fund Project of State Grid Hebei Electric Power Co. (kj2020-062).

References

[1] Fu Wenjie,Yang Di, Ma Hongming, et al. Short text classification method based on BTM and BERT[J]. Computer Engineering and Design, 2022, 43(12):3412-3427.

[2] Gan Yating, An Jianye, Xu Xue. Survey of Short Text Classification Methods Based on Deep Learning[J]. Computer Engineering and Applications, 2023: 1-13.

[3] Qijun Duan. Method of Short Text Classification based on TF-IWF Feature Selection[J]. International Journal of Social Science and Education Research,2021,4(4): 367-375.

[4] Wu Hanyu, Yan Jiang, Huang Shaobin, et al. CNN_BiLSTM_Attention Hybrid Model for Text Classification[J]. Computer Science, 2020, 47(S2): 23-27+34.

[5] Huang Jinjie, Lin Jiangquan, et al. Chinese Short Text Classification Algorithm Based on Local Semantics and Context[J]. Computer Engineering and Applications, 2021, 57(06): 94-100.

[6] X. Chen, P. Cong and S. Lv. A Long-Text Classification Method of Chinese News Based on BERT and CNN[J]. IEEE Access, 2022, 10: 34046-34057.

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

[8] Sun Hong, Chen Qiangyue. Chinese Text Classification Based on BERT and Attention [J]. Journal of Chinese Computer Systems:2021, 1-6

[9] Lv Xueqiang, Peng Chen, Zhang Le, et al. Text multi-label classification method incorporating BERT and label semantic attention[J]. Journal of Computer Applications,2022,42(01): 57-63.

[10] Cheng Siwei, Ge Weiyi, Wang Yu. BGCN: Trigger Detection Based on BERT and Graph Convolution Network[J]. Computer Science, 2021, 48(07): 292-298.