DOI: 10.19884/j.1672-5220.202101014

Long Text Classification Algorithm Using a Hybrid Model of Bidirectional Encoder Representation from Transformers-Hierarchical Attention Networks-Dilated Convolutions Network

ZHAO Yuanyuan(赵媛媛)[1, 2], GAO Shining(高世宁)[1, 2], LIU Yang(刘 洋)[1, 2], GONG Xiaohui(宫晓蕙)[1, 2*]

1 College of Information Science and Technology, Donghua University, Shanghai 201620, China

2 Engineering Research Center of Digitized Textile & Apparel Technology, Ministry of Education, Donghua University, Shanghai 201620, China

Abstract: Text format information is full of most of the resources of Internet, which puts forward higher and higher requirements for the accuracy of text classification. Therefore, in this manuscript, firstly, we design a hybrid model of bidirectional encoder representation from transformers-hierarchical attention networks-dilated convolutions networks (BERT_HAN_DCN) which based on BERT pre-trained model with superior ability of extracting characteristic. The advantages of HAN model and DCN model are taken into account which can help gain abundant semantic information, fusing context semantic features and hierarchical characteristics. Secondly, the traditional softmax algorithm increases the learning difficulty of the same kind of samples, making it more difficult to distinguish similar features. Based on this, AM-softmax is introduced to replace the traditional softmax. Finally, the fused model is validated, which shows superior performance in the accuracy rate and *F*1-score of this hybrid model on two datasets and the experimental analysis shows the general single models such as HAN, DCN, based on BERT pre-trained model. Besides, the improved AM-softmax network model is superior to the general softmax network model.

Key words: *long text classification*; *dilated convolution*; *BERT*; *fusing context semantic features*; *hierarchical characteristics*; *BERT_HAN_DCN*; *AM-softmax*

CLC number: TP391.1 **Document code**: A

Article ID: 1672-5220(2021)04-0341-10

Open Science Identity (OSID)

Introduction

Text classification is aimed at simplifying messy text data and summarizing information from unstructured data[1]. It is a basic task in natural language processing (NLP) and can be applied to sentiment classification, web retrieval, and spam filtering systems[2]. Specific classification rules are a necessary process for automatic text categorization, which mainly include text feature extraction and word vector representation.

For text feature extraction, experts have proposed a variety of methods, which can be summarized into the following: expert systems, machine learning, and deep neural networks, which are also the three main stages of NLP development. The expert system uses experts with relevant field expertise and experience to summarize rules and extract features for classification, which makes it difficult to deal with the flexible and changeable characteristics of natural language, and long-term dependence on manual feature extraction requires huge manpower. Machine learning algorithms[3-4] is a shallow feature extractor and this kind of feature engineering is based on manual extraction and is not able to automatically extract features from training sets.

However, in most of the above-mentioned feature extraction methods, high dimension and data sparseness result in poor performance[5]. With the rise and popularity of deep learning, neural networks have acquired excellent achievement in the field of image processing[6-7], and related scholars began to utilize deep learning[8-13] for NLP, which has been known as the feature extraction unit and has gained extraordinary accomplishments. The most representative neural network is convolution neural networks (CNN)[8] which is strong in feature learning, and it improves the feature extraction ability by modifying hyperparameters or increasing the number of layers of convolution, but at the same time facing the problems of a large amount of calculation and parameters adjusting. Dilated convolution network (DCN) is a variant network of CNN network. DCN is able to extract more global features with less parameter-adjusting works[14], but it often loses key information and context structure semantic information in obtaining global information. Attention mechanism can calculate the key information in characters

Received date: 2021-01-29

Foundation item: Fundamental Research Funds for the Central University, China (No. 2232018D3-17)

* Correspondence should be addressed to GONG Xiaohui, E-mail: xhgong@dhu.edu.cn

Citation: ZHAO Y Y, GAO S N, LIU Y, *et al.* Long text classification algorithm using a hybrid model of bidirectional encoder representation from transformers-hierarchical attention networks-dilated convolutions network [J]. *Journal of Donghua University (English Edition)*, 2021, **38**(4): 341-350.

and sentences[9]. Traditional attention mechanism usually performs on characters, but it is inadequate for the acquisition of semantic information, and afterwards Yang *et al.*[15] proposed a hierarchical attention neural network (HAN). HAN is composed with a two-level attention mechanism on characters and sentences, which cound effectively identify features, structural information and key value semantics. However, at the same time it lost its global features extraction and may generate partial semantic loss.

In the aspect of vector representation, unsupervised training is essential in vector representation of text, and the pre-trained CNN[16-17] are widely used to fine-tune the downstream tasks[18-19] gaining significant enlarged ability in feature extraction, transfer learning and dynamically fetch context semantics. The traditional models, such as fast text[20] and Glove[21], intend to obtain the semantic information of each word, discarding the semantic relevance with preceding texts, and is prone to the problems of dimension explosion and data sparseness[22]. Bidirectional encoder representation from transformers (BERT) is one of pre-trained word vector models that constructed with the n-layer transformer models with strong coding ability, and is able to calculate the semantic weight of each word with others in the sentences. Therefore, the pre-trained language model BERT is used as migration learning to fine-tune downstream tasks.

With the explosive growth of numbers of texts, a single classifier is not able to accomplish the tasks with high accuracy and precision, and many studies with mixed models have been proven more effective compared to single models in dealing with text classification problems[23-26]. A feature-fused HAN-DCN model was presented in this manuscript, the BERT model trained word vectors to initially understand the text semantics, HAN network obtained the structural dependency between word vectors, and the DCN extracted global and edge semantics in parallel. The features obtained from the two channels are spliced to be more efficient in improving the weight of the key information in the two levels of words and sentences, and extracting global semantic features as much as possible to improve the accuracy. Since softmax is aiming to maximize the probability of categorization by optimize the variances between different classes, and it is unable to minimize the differences within the same category, AM-softmax[27] is used to deepen the feature learning in improving the accuracy and efficiency of news text classification. The feasibility of the BERT_HAN_DCN model based on AM-softmax is verified through a series of experiments and it shows certain advances in improving generalization ability and model convergence speed.

1 Model Architecture

The entire architecture of this manuscript is indicated in Fig. 1. Firstly, the data is processed by BERT to initially get a rudimentary idea of the text so that we can obtain the dynamic semantic representation. After receiving the vector of the individual word in the long sentence, the digital vector is sent to the parallel network, which is composed by a three-layer DCN, which can acquire a larger receptive filed with fewer amount of calculation and HAN hybrid model to extract more abundant semantic information and contextual features information. In the relevant image processing, the mesh effect appears in the dilated convolution, resulting in the loss of characteristic information[28-29]. Therefore, in this network design, a three-layer dilated convolution is adopted to overcome the influence of the mesh effect, and the coefficients of expansion of each layer are set to 1, 3, and 5, respectively. Thus, the feature representation of the text is formed by combing the feature information of these two parts. In the end, the softmax function is used to normalize and classify the output probability according to the probability size. The mixed model architecture is shown in Fig. 1.

1.1 Input of representation layer

BERT could obtain dynamic and nearly comprehensive semantic information of the text. The BERT model uses a transformer with a bidirectional structure to fuse left and right characters to obtain contextual semantics, complete the two tasks of masking language model (MLM) and next sentence prediction (NSP) at the same time, and conduct joint training to obtain the vector representation of words and sentences. BERT's embedding layer consists of token embedding (vector representation of words), segment embedding (vector representation of two sentences in a sentence pair, similarity of the sentence pair) and position embedding (learning the order properties of the sentence) to convert Chinese characters into input vectors W_1, W_2, $\cdots$, W_n, and the model can dynamically generate the context semantic representation of words by bidirectional transformer structure[30] to perform the two tasks mentioned above (MLM and NSP) as shown in Fig. 2. The final transformer output of the hidden layer vector with semantic information is avilable from the self-attention layer, the remaining connection and the normalization layer, and the obtained output is the superposition of the character-level vector. The output layer vectors processed by BERT are E_1, E_2, $\cdots$, E_n, which are obtained by multi-layer transformers. In this experiment, BERT_BASE_CHINESE model is used, which is composed by a 12 layers-multi-head attention mechanism transformer.

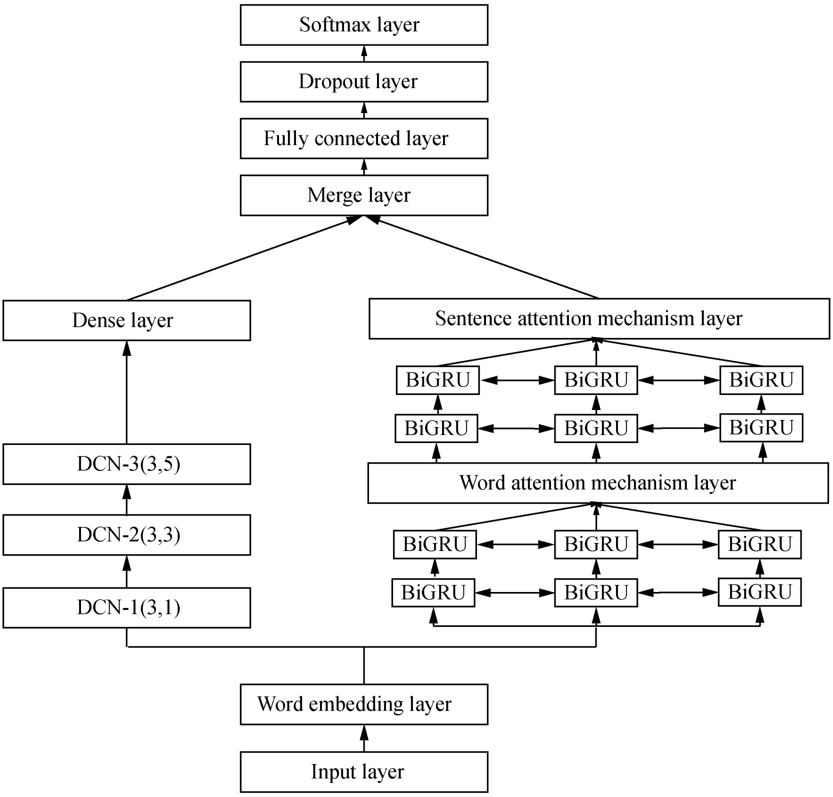


Fig. 1 Overall framework of BERT_HAN_DCN

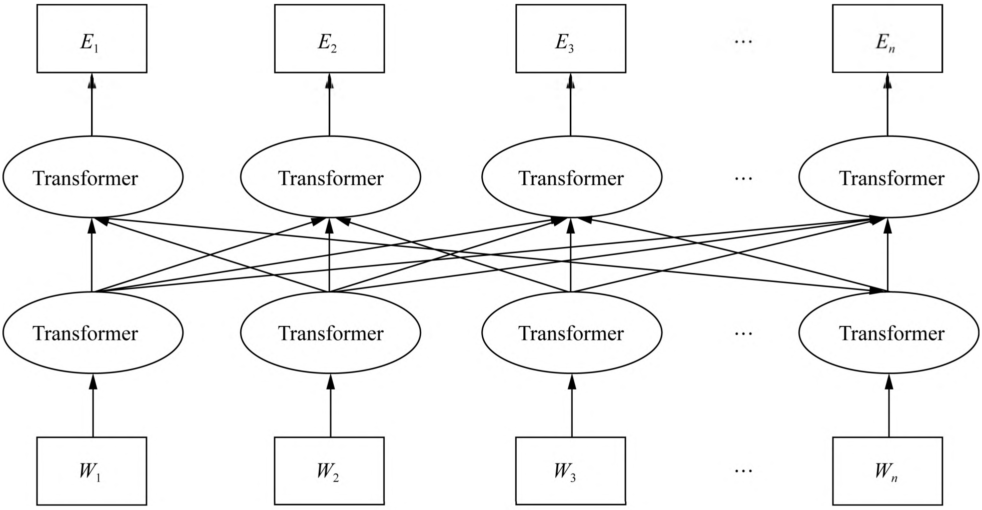


Fig. 2 BERT model structure

1.2 HAN layer

As illustrated in Fig. 3, the HAN model is composed by Chinese character and sentence level attention network, and this hierarchical structure is in line with people's habitual thinking of understanding articles. In essence, each layer in attention network is composed by two layers of BiGRU, which has the advantage of serialization learning text features in the dotted box in Fig. 3. Considering the hierarchical structure of the network, it is necessary to set the fixed length of each sentence when dividing sentence attention. Thus, in this manuscript, we maximize the characters of the longest sentence in the article as 256, and each sentence is divided into segments with 16 characters. HAN is composed of four parts: word encode, word attention, sentence encode and sentence attention, which will be explained the calculation process in detail.

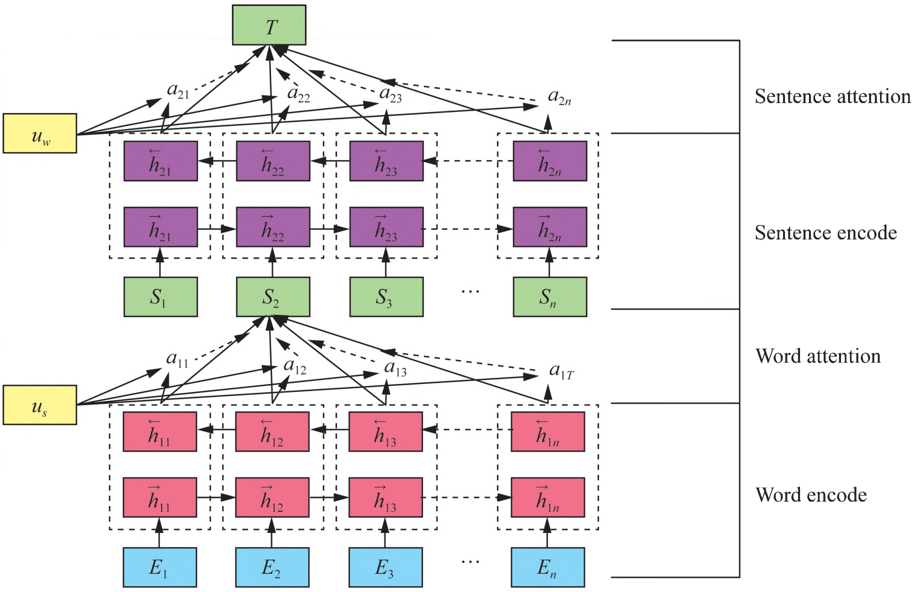


Fig. 3 HAN model structure

(1) Word encode

In this part, the embedding layer vector is word-encoded. The vectors are initialized and then used as the input of the two-layer BiGRU. The specific conversion method is shown as

$$x_{in} = E_e E_n,\ n \in [1,\ t], \tag{1}$$

$$\overrightarrow{h}_{1n} = \overrightarrow{GRU} x_{in},\ n \in [1,\ t], \tag{2}$$

$$\overleftarrow{h}_{1n} = \overleftarrow{GRU} x_{in},\ n \in [1,\ t], \tag{3}$$

where E_e is the random initialization weight matrix, x_{in} is the random initialization matrix, $\overrightarrow{h}_{1n}$ is the forward hidden state of x_{in} output through the BiGRU layer, and $\overleftarrow{h}_{1n}$ is the reverse hidden state of x_{it} output through the BiGRU layer.

(2) Word attention

The splicing vectors h_{11}, h_{12}, …, h_{1n} of the forward hidden state and the reverse hidden state are used as the overall expression of the word. In this part, calculate the size of the attention weight matrix of each word in the sentence. The calculation method is

$$u_{1n} = \tanh(w_s h_{1n} + b),\ n \in [1,\ t], \tag{4}$$

$$\alpha_{1i} = \frac{\exp(u_{1n}^{n}\ u_s)}{\sum_n \exp(u_{1n}\ u_s)},\ n \in [1,\ t] \tag{5}$$

$$S_i = \sum_n \alpha_{1i} \overleftarrow{h}_{1n},\ n \in [1,\ t], \tag{6}$$

where tanh is activation function, u_{1n} is a hidden representation of $\overleftarrow{h}_{1n}$, u_s is a randomly initialized word vector, and α_{1i} is the probability of similarity between u_{1n} and S_i is the importance of each word. The purpose of adding a mechanism in this step is to discover the significant meaningful sentences in the text.

(3) Sentence attention

Similarly, the sentence encode layer is the same as that in the word encode layer, $\overleftarrow{h}_{2n}$ represents the reverse hidden state of s_i output through the first BiGRU layer with attention, and the calculation of sentence attention formula is shown as

$$u_{2n} = \tanh(w_w \overleftarrow{h}_{2n} + b),\ n \in [1,\ t], \tag{7}$$

$$\alpha_{2i} = \frac{\exp(u_{2n}^{n}\ u_w)}{\sum_n \exp(u_{2n}\ u_w)},\ n \in [1,\ t], \tag{8}$$

where u_{2t} is a hidden representation of $\overleftarrow{h}_{2n}$, u_w is a randomly initialized sentence vector, and α_{2i} is the probability of similarity between u_{2n}.

The purpose of adding a mechanism in this step is to discover the significant meaningful words in the sentence. We can get the final output of the HAN network as illustrated in Eq. (9), in which the vector ω_1 is the significant local characteristic of the mixed neural network, which is defined as

$$\omega_1 = \sum_n \alpha_{2n} h_{2n},\ n \in [1,\ t], \tag{9}$$

where ω_1 is the document vector as well as the final features extracted by HAN, which sums up all the information of the sentence in long text.

1.3 Dilated convolutional networks layer

As shown in Fig. 2, the DCN layer is composed of three hollow convolutional blocks with the same structure, and the input of each dilated convolutional layer is the output of the earlier layer. Changing the dilated convolution rate of each layer allows the receptive field of the convolutional layer to quickly cover all input data. As the input expansion rate of each layer increases, the obtained feature information increases exponentially.

DCN and HAN are parallel network structures, taking the output of the embedding layer, which initialized by BERT as the input, and the input of each word in the sentence are $E_i \in R^{B \times N \times D}$, where B is batch_ size which is set to 64, N is the number of words, and D is the word vector dimension of BERT output. The feature extraction of the input text sentence by dilated convolution is completed by setting the filter size. The convolution calculation is shown as

$$c_i = f(\omega \cdot E_{i:i+k+(k-1)(r-1)} + b), \tag{10}$$

where f is a non-linear function, ω is the random initialization weight matrix the convolution kernel, k is the size of the convolution kernel, and r is the hole rate of the hole convolution; $E_{i:i+k+(k-1)(r-1)}$ is $i:i+k+(k-1)(r-1)$ the sentence vector composed of i to $i+k+(k-1)(r-1)$, and b is the bias term.

Therefore, after the feature extraction of the dilated convolutional layer, the final vectors obtained are C. The concrete vector representation of C is shown as

$$C = [c_1, c_2, \cdots, c_{i+k+(k-1)(r-1)}]. \tag{11}$$

The output of HAN is going to be serialized continuous vectors, which needs to keep the dimensions consistently. Connect the vector obtained from the three layers dilate convolution networks and convert the vector into a feature matrix w_2, which is as shown in

$$w_2 = [C_1, C_2, \ldots, C_i], \ i \in [1, n], \tag{12}$$

where C_i is the feature matrix of the output of dilated convolutional neural networks.

1.4 Classification layer

The classification layer is composed of the following four parts: feature fusion layer, fully connected layer, dropout layer and softmax layer. It consists of a simple softmax classifier (at the top of HAN and DCN) to calculate conditional probability distributions on predefined classification tags. Using Keras's add function at the model fusion layer, we can get the merge layer vector ω, shown as

$$\omega = w_1 \oplus w_2, \tag{13}$$

where w_1 and w_2 represent the features output vectors of HAN and DCN respectively, and $\oplus$ represents a splicing operation. After realizing merge layer operation, the obtained feature vectors are combined. Then extracting the feature vector again, each input unit of the fully connection layer represents the value of each feature vector. In order to avoid overfitting of the model, we use the dropout mechanism. The final feature representations are obtained from the dropout layer, and these feature representations are classified by softmax classification algorithm. The classification algorithm calculates the probability of ω into category z, and the concrete calculation formula is shown as

$$P(y^{(i)} = z \mid \omega^{(i)}; \theta) = \frac{\exp(\theta_z^{\mathrm{T}} x^{(i)})}{\sum_{n=1}^{k} \exp(\theta_n^{\mathrm{T}} x^{(i)})}, \tag{14}$$

where θ represents all parameters in training, which is the output vector representation of the dropout layer, k is the number of classes and this manuscript is all about eight classification tasks, θ_n is the nth column of the dropout layer, and $\theta_z^{\mathrm{T}} x^{(i)}$ is the logarithm of the object for the ith sample. Finally, the item with the largest probability value in P is selected as the final predicted classification label.

2 Experiments and Results

In this section, for verifying the effectiveness of BERT _ HAN _ DCN model, we use two real-world experimental datasets. We which are extracted the portion from SogouCS and THCNews datasets, explicate the details of the experiment, evaluate the performance of the hybrid model, and analyze the experimental results.

2.1 Experimental datasets

The datasets used in this experiment are Chinese text classification datasets launched by the NLP Laboratory of Tsinghua University and Sogou labs. The detailed data amount of the train group, the validation group and the test group are shown in Table 1.

Table 1 Details of the text classification datasets

Parameter name	SogouCS	THCNews
Field	News	News
Classification	8	8
Train	25 110	25 968
Validation	3 137	3 246
Test	3 137	3 246

2.2 Multi-classification evaluation index

On the course of training process of the text classifier, it is indispensable to select appropriate criteria to evaluate the ability of the classifier. The confusion matrix is shown in Table 2 and there are four commonly used criteria in the field of NLP: precision (P), accuracy (A), recall (R), and $F1$-score ($F1$).

Table 2 Confusion matrix

Classification result	Actually positive	Normal negative
Classified as positive	a	b
Classified as negative	c	d

(1) Accuracy(A)

$$A = \frac{a + d}{a + b + c + d}, \tag{15}$$

where A is measuring the ability of the classifier to distinguish the whole data set, the higher the value of A represents the better classification ability the model has.

(2) $F1$-score ($F1$)

$$F1 = \frac{2 \times P \times R}{P + R}, \tag{16}$$

where P is shown in

$$P = \frac{a}{a + b}, \tag{17}$$

and R is shown in

$$R = \frac{a}{a + c}. \tag{18}$$

$F1$ is a comprehensive index which is the harmonic evaluation value of precision and recall. It can be seen $F1$ combines the results of P and R , and when $F1$ gets closer to 1, it can indicate that the model method is more effective.

2.3 Main initialization hyperparameters

In order to train a better classification model, we should set the appropriate hyperparameters settings of the model. The hidden vector dimension of BiGRU and DCN models are respectively set to 64 and 128, the size of batch is 64, the dropout rate of BiGRU is set to 0.1, the maximum input length of data set is set to 256, and the learning rate is 0.000 05. The other main initialization hyperparameters settings of this experiment are shown in Table 3.

Table 3 Main initialization hyperparameters

Hyperparameters	Practical purpose	Value
Hidden_size	Hidden neurons of BERT	768
Epoch	Number of iterations	10
Kernel_size	Feature extraction	3
Dilate rate	Expansion rate	(1, 3, 5)

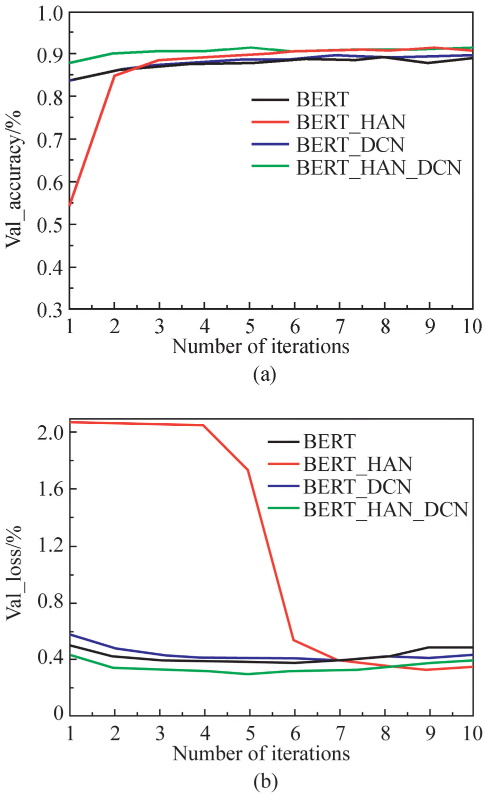
Using optimizer Adam[31] to update network weights and cross-entropy cost function is used for calculating loss. In addition, early stopping is used to prevent over fitting. After multiple training processes in the models, it is found that 3 is the most suitable parameter for all experimental models. Complicated neural networks trained on small data sets often result in overfitting[32–33]. Because of the fewer data sets in this experiment, a certain dropout rate is adopted to prevent the overfitting of the model. Consequently, five groups of experiments were designed to explore the influence of dropout rate on the model effect and the optimal parameters were suitable for this fusion model. When we change the dropout rate, every 0.1 change has an impact on the accuracy of the model. Finally, we found the most appropriate parameter dropout for SogouCS dataset is set to 0.6, while THCNews dataset is set to 0.8.

2.4 Analysis of experimental results

2.4.1 Results of hybrid model

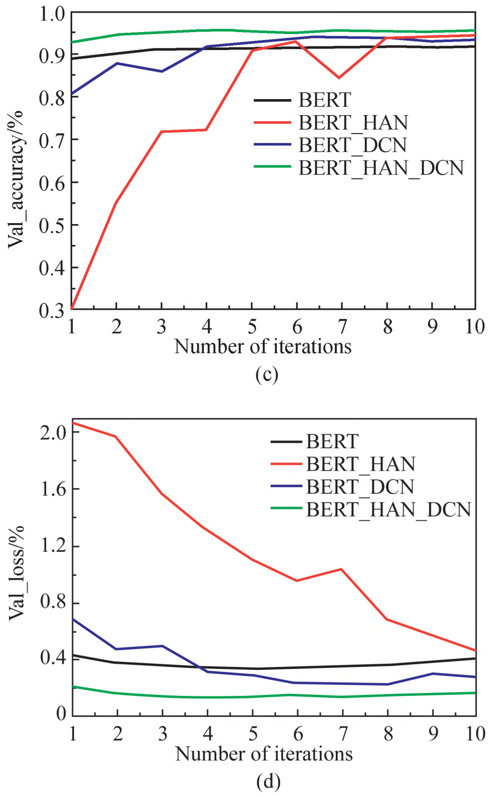
This manuscript focuses on two kinds of news data sets. A total of three groups of comparative experiments are designed, which are using the standard BERT to connect the fully connection layer directly, and using the BERT as the embedded representation layer, HAN and DCN as the feature extraction layer respectively. The accuracy and loss rate of the two data sets during the training process are plotted separately, as shown in Fig. 4. The models are trained for 10 epochs on two datasets. Over the course of training data, it can be seen from Fig. 4 that the BERT_HAN model plays a significant role in improving accuracy. However, the model tends to be unstable in the first few iterations. The reason for this phenomenon is the complex structure of HAN model network. In the early stage of learning, the error is relatively large, and some important features may be lost when focusing on local important features. With the constant updating of parameters and BERT_HAN's strong learning ability, the accuracy and stability of prediction are constantly improved. The BERT_DCN model is more stable than BERT model in both data sets. In addition, the accuracy of data set SogouCS and THCNews improved by 2.89% and 2.03% respectively compared with the BERT model.

From the experiment results, in SogouCS and THCNews, BERT_HAN_ACN model achieved accuracy value of 91.42% and 95.66% respectively and the loss rate of 39.95% and 17.83% respectively. The accuracy of the BERT_HAN_DCN model in the verification set is higher than that of other models, and it is more stable in the training process than the other models. Compared with other groups of models are showed the best effect which has improved considerably and shows that the designed model fusion is feasible, which can extract deep characteristics of long text and improve the effect of news text classification model.



(a)

(b)



(c)

(d)

Fig. 4 Training performance comparison between the presented model and other basic models: (a)−(b) training curves of verification accuracy and loss of SougoCS; (c)−(d) training curves of verification accuracy and loss of THCNews

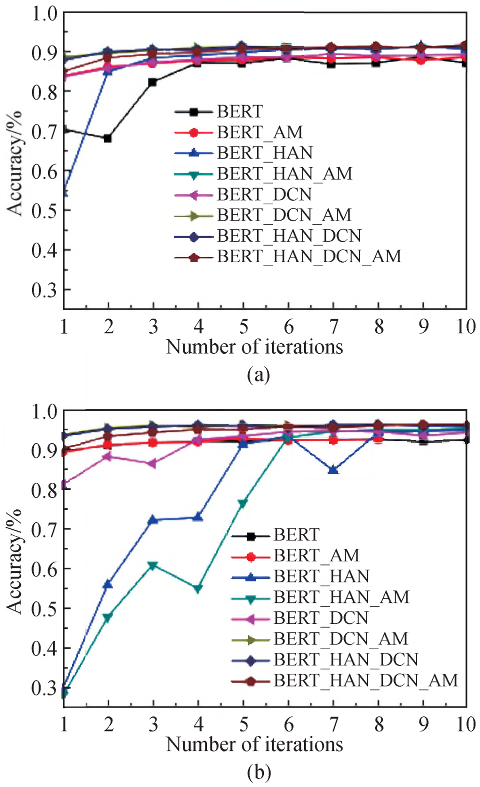
2.4.2 Impact of AM-softmax

AM-softmax has achieved remarkable results in the field of face recognition. Unlike softmax, AM-softmax can reduce the probability of correct label and increase the effect of loss, which is more helpful to the aggregation of the same class. The specific AM-softmax is shown as

$$L' = \frac{e^{s*(\cos\theta_{yj}-m)}}{e^{s*(\cos\theta_{yj}-m)} + \sum_{i,\, i\neq j}^{N} e^{s*\cos\theta_i}}, \tag{19}$$

where $\cos\theta_{yj}$ is to calculate x_j in the category y_j region; m is the area between categories which is at least m apart. The value of m here is set to 0.35 which needs to consider whether there is a clear boundary between the distribution of data in the real scene. The cosine value is between [0, 1], which is too small and cannot effectively distinguish the difference. After increasing s times to improve the difference of distribution and s here is set to 30. And with the increase of the number of training epochs, the accuracy of validation sets of different models changed as shown in Fig. 5.

From the accuracy of the verification sets, it can be concluded that after 5−6 times of model training about two datasets, the mixed model BERT_HAN_DCN which based on AM-softmax tends to be stable and finally achieves higher accuracy.



(a)

(b)

Fig. 5 Training performance comparison between different models: (a) training accuracy curves of SougoCS based on AM-softmax models and the original models; (b) training accuracy curves of THCNews based on AM-softmax models and the original models

The training models were verified by the validation set after 10 epochs. The precision, recall, *F*1 value and accuracy of the 8 categories of the two data sets are obtained respectively as shown in Tables 4−5.

As shown in Tables 4−5, the models using AM-softmax as loss function have slight improvement in both accuracy rate and *F*1 value compared with the original models. Although the improvement effect is small, it also proves that changing the way of calculating loss is also a way to improve the feature extraction ability of the training model. Finally, for SogouCS dataset, we find that the final *F*1-score and accuracy of the hybrid model are respectively increased by 0.56% (from 91.42% to 91.98%) and 0.55% (from 91.42% to 91.97%). For THCNews dataset, we find that *F*1-score and accuracy of the hybrid model are respectively increased by 0.34% (from 95.69% to 99.06%) and 0.3% (from 95.66% to 95.69%).

Table 4 Model comparison result on SogouCS dataset

Model	Precision/%	Recall/%	*F*1/%	Accuracy/%
BERT	88.70	89.04	88.77	88.69
BERT_AM	89.66	89.75	89.68	89.61
BERT_HAN	90.98	91.18	91.00	90.95
BERT_HAN_AM	91.50	91.42	91.43	91.42
BERT_DCN	89.88	89.90	89.87	89.84
BERT_DCN_AM	91.62	91.65	91.60	91.58
BERT_HAN_DCN	91.50	91.54	91.42	91.42
BERT_HAN_DCN_AM	91.97	92.08	91.98	91.97

Table 5 Model comparison result on THCNews dataset

Model	Precision/%	Recall/%	*F*1/%	Accuracy/%
BERT	92.09	92.09	92.06	92.03
BERT_AM	92.74	93.21	92.87	92.70
BERT_HAN	94.45	94.62	94.52	94.49
BERT_HAN_AM	95.01	95.12	95.04	94.98
BERT_DCN	94.43	94.35	94.35	94.36
BERT_DCN_AM	95.33	95.64	95.45	95.44
BERT_HAN_DCN	95.75	95.66	95.69	95.66
BERT_HAN_DCN_AM	95.90	96.22	96.03	95.96

2.4.3 Time complexity comparison experiment

Under the same parameter settings, the algorithm running time of 8 different modules was compared, and the effectiveness of the algorithm was verified. The time complexity comparison experiment results are shown in Table 6.

Table 6 Time complexity comparison

Model	Time complexity	
	SogouCS/s	THCNews/s
BERT	76	79
BERT_HAN	277	289
BERT_DCN	112	115
BERT_HAN_DCN	306	323
BERT_AM	76	78
BERT_HAN_AM	271	282
BERT_DCN_AM	102	105
BERT_HAN_DCN_AM	300	312

From Tables 4–6, experimental results show that the mixed model has higher time complexity. However, the accuracy and *F*1 are much better. For SogouCS and THCNews datasets, the average calculation time per epoch of the hybrid model are 230 s and 244 s longer than BERT, respectively, but the accuracy is improved. It is obvious that the addition of hierarchical attention mechanism increases the complexity of the computing complexity but effectively improves the accuracy of the model. The calculation time of all AM-softmax-based models is less than that of the original models. For the SogouCS dataset, the average calculation time of each round of the hybrid model is reduced by 6 s, and for the THCNews dataset, the average calculation time of each round of the hybrid model is reduced by 11 s. This proves that the calculation of changing the loss improves the convergence speed of the model to a certain extent and slightly reduces the complexity of the model.

3 Conclusions

This manuscript adopts the BERT _ HAN _ DCN model of the composite network and applies it to the task of Chinese long text classification. Compared with the single BERT model, BERT_HAN, BERT_DCN, the accuracy and *F*1 value of the model are the highest. The results show that the fusion of HAN and DCN is effective and can learn deep features and contextual information in long text.

In addition, by improving the loss function, the accuracy and *F*1 of the single model and the mixed model are improved and relatively reduced training time which proves that the mixed model can be better applied to Chinese text classification tasks. This also shows that in the process of model training, not only the ability of feature extraction and word vector transformation, but also the impact of loss function on model accuracy should be paid attention to.

However, a more complex hybrid model requires more network parameters, which requires more computing power and longer training time. In the following research, we intend to further optimize and improve the details of the algorithm and we will improve this work by building a larger dataset.

References

[1] ZHENG J M, CAI F, SHAO T H, *et al*. Self-interaction attention mechanism-based text representation for document classification [J]. *Applied Sciences*, 2018, **8**(4): 613.

[2] COLLOBERT R, WESTON J, BOTTOU L, *et al*. Natural language processing (almost) from scratch [J]. *Journal of Machine Learning Research*, 2011, **12**(1): 2493-2537.

[3] PU Q, YANG G W. Short-text classification based on ICA and LSA[C]// International Symposium on Advances in Neural Networks Chengdu, China. Berlin: Springer, DBLP, 2006: 265-270.

[4] CHEN Z G, SHI G, WANG X J. Text classification based on naive bayes algorithm with feature selection [J]. *International Journal on Information*, 2012, **15**(10): 4255-4260.

[5] WANG K, LIU B S. A review of text classification research [J]. *Data Communication*, 2019, **9**(3): 37-47. (in Chinese)

[6] CHEN T, LU S J, FAN J Y. SS-HCNN: Semi-supervised hierarchical convolutional neural network for image classification [J]. *IEEE Transactions on Image Processing*, 2019, **28**(5): 2389-2398.

[7] OPPENHEIM D, SHANI G, ERLICH O, *et al*. Using deep learning for image-based potato tuber disease detection [J]. *Phytopathology*, 2018, **109**(6): 1083-1087.

[8] NAL K, EDWARD G, PHIL B. A convolutional neural network for modelling sentences [C]// Proceedings of the 52nd Annual Meeting of Computational Linguistics, 2014: 655-665.

[9] SHEN T, ZHOU T, LONG G, *et al*. DiSAN: Directional self-attention network for RNN/CNN-free language understanding [C]// Proceedings of the AAAI Conference, 2018, **32**(1).

[10] KIM Y. Convolutional neural networks for sentence classification [C]// Conference on Empirical Methods in Natural Language Processing. Stroudsburg: ACL Press, 2014: 1746-1751.

[11] MIYATO T, DAI A M, GOODFELLOW I. Adversarial training methods for semi-supervised text classification [J]. arXiv: 1605.07725, 2017: 1-12.

[12] SUN H, CHENG H Y. Chinese text classification combining BERT word embedding and attention mechanism [J]. *Small Microcomputer System*, 2021: 1-6. (in Chinese)

[13] LAI S W, XU L H, LIU K, *et al*. Recurrent convolutional neural networks for text classification [C]// Proceedings of the 29th AAAI Conference on Artificial Intelligence, 2015: 2267-2273.

[14] WANG P Q, CHEN P F, YUAN Y, *et al*. Understanding convolution for semantic segmentation [C]// 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), NY, USA. 2018: 1451-1460.

[15] YANG Z C, YANG D Y, DYER C, *et al*. Hierarchical attention networks for document classification [C]// Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Dieg, California. 2016: 1480-1489.

[16] DEVLIN J, CHANG M W, LEE K, *et al*. BERT: pre-training of deep bidirectional transformers for language understanding [C]// Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, 2019: 4171-4186.

[17] KAHYUN, LEE, MICHELE, FILANNINO, *et al*. An empirical test of GRUs and deep contextualized word representations on de-identification [J]. *Studies in Health Technology and Informatics*, 2019, **264**: 218-222.

[18] HOWARD J, SEBASTIAN R. Universal language model fine-tuning for text classification. [C]// Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia. 2018: 328-339

[19] ZHANG X C, DAI X R, LIU L, *et al*. Chinese short text classification model incorporating multi-head self-attention mechanism [J]. *Computer Application*, 2020, **40** (12): 3485-3489. (in Chinese)

[20] LE Q, MIKOLOV T. Distributed representations of sentences and documents [J]. *International Conference on Machine Learning*, 2014: 1188-1196.

[21] PENNINGTON J, SOCHER R, MANNING C. Glove: global vectors for word representation [C]// Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, Doha, Qatar. 2014: 1532-1543.

[22] JOULIN A, GRAVE E, BOJANOWSKI P, *et al*. Bag of tricks for efficient text classification[C]// Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, Valencia, Spain. 2017: 427-431.

[23] ZHOU C, SUN C, LIU Z, *et al*. A C-LSTM neural network for text classification [J]. *Computer Science*, 2015, **1**(4): 39-44.

[24] HOU X L, LI X, CHENG Y P. Short text classification model based on hybrid multi-neural networks[J]. *Computer System Application*, 2020,

29(10): 9-19. (in Chinese)

[25] LI C B, ZHAN G H, LI Z H. News text classification based on improved Bi-LSTM-CNN [C]//2018 9th International Conference on Information Technology in Medicine and Education (ITME), Hangzhou, China. 2018: 890-893.

[26] HUANG H, JING X Y, WU F, *et al.* DCNN-BiGRU text classification model based on BERT embedding [C]//2019 IEEE International Conferences on Ubiquitous Computing & Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (SmartCNS), Shenyang, China. 2019: 632-637.

[27] WANG F, CHENG J, LIU W, *et al.* Additive margin softmax for face verification [J]. *IEEE Signal Processing Letters*, 2018, **25**(7): 926-930.

[28] YU F, KOLTUN V, FUNKHOUSER T. Dilated residual networks [J]. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017: 472-480.

[29] WANG Z Y, JI S W. Smoothed dilated convolutions for improved dense prediction [C]// KDD 18: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Date Mining, London, UK. 2018: 2486-2495.

[30] VASWANI A, SHAZEER N, PARMAR N, *et al.* Attention is all you need [C]// Proceedings of the 31st International Conference on Neural Information Processing Systems, California, USA. 2017: 5998-6008

[31] CHANG Z H, ZHANG Y, CHEN W B. Effective Adam-Optimized LSTM neural network for electricity price forecasting [C]//2018 IEEE 9th International Conference on Software Engineering and Service Science, Beijing, China. IEEE, 2018: 245-248.

[32] SRIVASTAVA N, HINTON G, KRIZHEVSKY A, *et al.* Dropout: a simple way to prevent neural networks from overfitting [J]. *Journal of Machine Learning Research*, 2014, **15**(1): 1929-1958.

[33] HINTON G E, SRIVASTAVA N, KRIZHEVSKY A, *et al.* Improving neural networks by preventing co-adaptation of feature detectors [J]. *Computer Science*, 2012, **3**(4): 212-223.