International Journal of Software Innovation
Volume 11 • Issue 1

Sentiment Analysis of Hybrid Network Model Based on Attention

Wenqian Shang, Communication University of China, China*
Hongzhan Zhen, Communication University of China, China
Wanyu Zhang, Communication University of China, China

ABSTRACT

The existing text sentiment analysis models based on deep learning and neural network usually have problems such as incomplete text feature extraction and failure to consider the impact of key information on text sentiment tendency. Based on the parallel hybrid network and the two-way attention mechanism, an improved text sentiment analysis model is proposed. The model first takes the word vector trained by the BERT language model as the input, and then extracts the global and local features of the context simultaneously through the parallel hybrid neural network constructed by the Convolution Neural Network (CNN) and The Bidirectional Gated Recurrent Unit (BiGRU), so as to improve the feature extraction ability of the model. It also integrates the dual-way attention mechanism to strengthen the key information in the global feature and local feature, and the feature vectors obtained by feature fusion are used for sentiment analysis.

KEYWORD

Attention Mechanism, Deep Learning, Neural Network, Sentiment Analysis

INTRODUCTION

With the rapid development of the Internet, the era of big data has arrived. More and more people express their opinions and share their feelings online, resulting in massive data. Because of the text creator's position and preferences, the created text will have different emotional colors, showing different positive or negative attitudes. Making full use of and analyzing the user comments on such software, digging into the text content, and extracting deep information from the text will play a certain guiding role in the consumption choice of users, the decision-making of businesses and enterprises, and the monitoring of public opinions.

Sentiment analysis in the field of natural language processing (NLP) is an important branch, which is the application of natural language processing and text mining technology. Sentiment analysis emphasizes the subjectivity of color with emotional text analysis, processing, and extraction (Pang et al., 2008). It is used for feature extraction of text content, so as to help users quickly understand

DOI: 10.4018/IJSI.327364 *Corresponding Author

the emotional tendency of corresponding text. There are two kinds of sentiment analysis techniques widely used at present. The first one is the sentiment analysis of the sequence based on a pre-prepared sentiment dictionary (Zhang et al., 2018). According to the sentiment dictionary, different sentiment scores are assigned to the word segmentation results of the text content. Finally, the corresponding sentiment categories are output according to the scores. This method is relatively simple, but it requires a large sentiment dictionary and too much domain knowledge, and the data is difficult to collect and does not have good mobility. Thus, the effect is not very good. The second one is to use the traditional machine learning (2019) method to carry out a sentiment analysis of the text. This method needs to collect and label the features manually and then use machine learning algorithms such as decision trees and support vector machines for sentiment analysis. Although machine learning algorithms show good performance in sentiment analysis tasks, it is impractical to collect, process and label large texts manually. Therefore, it has become an urgent demand for people to use computers to automatically process and analyze data and quickly obtain valuable content. The development of deep learning has made it possible to use machines to automatically extract text features without having to annotate them manually. Compared with traditional machine learning, deep learning can spontaneously capture the deep information of text at the level of natural language, which significantly improves the accuracy and efficiency of text sentiment analysis.

With the wide application of deep learning in the field of sentiment analysis, many new neural network models have been proposed and improved, which show good results in the task of sentiment analysis. In recent years, some scholars have proposed the mechanism of Attention and applied it to the task of text sentiment analysis (Niu et al., 2021). The attention mechanism mimics the biological visual system, being able first to notice the most valuable features of data and extract the more critical information about the data. In order to capture the more emotionally inclined words in the text, this paper introduces the attention mechanism into the sentiment analysis task.

The innovations and contributions of this paper are as follows:

- A hybrid neural network sentiment analysis model (BCBA) based on the attention mechanism was constructed. In the text preprocessing stage, the BERT model is used to input the text word vector output produced by the BERT model into the hybrid neural network of CNN and BiGRU and integrate this vector into the attention mechanism. The model can better extract the features of the text and explore the emotional attributes.
- The corresponding benchmark model is established. In order to compare and analyze the mixed network model in this paper, the relevant benchmark model is set up. The language models used in the text preprocessing stage of model training are all BERT models, and the benchmark models are as follows: the CNN-fully connected-attention, CNN-BiGRU, CNN-BiLSTM-attention, CNN-attention, and BiGRU-attention models. We provide comparative experiments for the models in this paper.

RELATED WORK

Convolution neural networks (CNNs) are originally used for image processing after improvement and promotion in the field of natural language processing. In 2014, Kim et al. (2014) first proposed to take the pre-trained word vector as input and use a CNN to classify text emotions. In a traditional neural network model, two adjacent layers are fully connected; thus, a large number of parameters need to be trained, and the training efficiency is not high. After the introduction of CNNs, the key features of text can be identified and extracted in the convolution layer and pooling layer, reducing the number of parameters to be trained and avoiding overfitting because the key features are retained. Kim et al. used a CNN to carry out text classification tasks on multiple data sets and achieved good results. Therefore, some scholars have proposed a new CNN model successively and applied it to text classification tasks. Nal Kalchbrenner et al. (2014) proposed an algorithm to improve the pooling layer of a CNN, called dynamic convolutional neural networks (DCNN), which replaces the max pooling algorithm of the CNN pooling layer with the *k*-max pooling algorithm to dynamically select the number of features to be extracted in the pooling layer, retains certain text location features, and achieves good results. However, CNNs can only mine the local features of text, and the capture effect of long-distance dependence is not good enough. Therefore, some scholars have successively applied RNNs to supplement CNNs in text sentiment analysis.

Compared with traditional neural networks, an RNN (recurrent neural network) has a memory function, which makes up for the deficiency that a CNN cannot capture the correlation between long-distance contexts to some extent. Mikolov et al. (2013) proposed the use of an RNN for text classification. The output of an RNN at the current moment depends on the input of the current moment and the hidden layer state of the previous moment, so it can store the sequence information of the previous moment and learn the context of the text. However, an RNN is weak in processing long sequences, and the words input later have more advantages than the words before. Moreover, it can only process one-way sequences, and it is prone to gradient explosion and gradient disappearance in the training process.

In order to make up for the deficiency of RNNs, Hochreiter and Schmidhuber (1997) proposed a variant of RNNs, i.e., the LSTM (long short-term memory) model and optimized RNN. Based on RNNs, LSTM introduces the idea of the gate, which allows for the selection and retention of more important information through gate control units, rather than placing more emphasis on closer information, as an RNN does. Therefore, LSTM enhances the ability to extract and process long sequence information to a certain extent, while effectively mitigating the loss of gradients. Zhu et al. (2015) described a method of sequence modeling using LSTM, that is, text is divided into word sequences for sentiment classification, which achieved good results. A single LSTM can only extract the sequence information in one direction and cannot make full use of the context of the text. Therefore, some scholars proposed a bidirectional LSTM model to extract the sequence content from both directions and carry out feature fusion for sentiment analysis of the text. Liu et al. (2019) used BiLSTM model and integrated attention mechanism to effectively utilize contextual semantic information of text for sentiment analysis. However, LSTM requires too many parameters and complex structure, so training requires a lot of time and memory space. The threshold cycle unit method proposed by Cho et al. (2014; GRU) is a variant of LSTM. GRU also uses a gate structure but simplifies gate structure based on LSTM, with less computation and faster convergence, so the training efficiency is effectively improved and widely used. The bi-threshold cyclic unit (Bi-GRU) is a neural network connected by the output state of a forward propagating GRU and a backward propagating GRU. It can extract text features from both directions and effectively capture the global semantic information of text.

With the application of deep learning in the field of sentiment analysis, sentiment analysis of text based on the attention mechanism has become a hot research topic. The idea of the attention mechanism is to find the information that is more critical to the current task in a lot of information, so that it can play an important role in text feature extraction. Bahdanau et al. (2015) applied the attention mechanism to natural language processing tasks, weighted input features by the attention mechanism to measure the contribution of current input features to current recognition and achieved a good model effect. Since then, the attention mechanism has been widely used by text analysis scholars. Liu et al. (2019) proposed a hybrid network model based on the attention mechanism and used CNN and BiGRU for text sentiment analysis, which made great progress in accuracy compared with previous hybrid neural network models. Yang et al. (2016) proposed a hierarchical attention mechanism, which uses the attention mechanism at the word and sentence levels, respectively, and considers the importance of keywords and key sentences in the text, which is suitable for the classification and processing of long text.

The pretraining models used in previous NLP missions were usually Word2Vec and GloVe models. Although text could be converted into vectors, the conversion was static and did not take into account that a single word might mean different things in different contexts. Based on this, Google proposed a language model BERT (Devlin et al., 2019), which can be used to solve the NLP problem. BERT is a bidirectional deep pretraining language representation method. Compared with Word2Vec and other traditional pretraining methods, the vectors generated by BERT are dynamic, that is, the same word may correspond to different vectors in different contexts. The BERT model is used to process the text input, extract the text information from two directions at the same time, and convert the text into a vector for the downstream task of sentiment classification. BERT has been continuously applied to natural language processing-related tasks and shows good performance. Based on the above description, in order to build a better sentiment analysis model, this paper studies how to apply a mixed neural network to the attention mechanism in a sentiment classification task.

RELEVANT THEORETICAL BASIS OF SENTIMENT ANALYSIS

In order to build a good sentiment analysis model, this paper studies how to apply a mixed neural network to the attention mechanism in accomplishing a sentiment classification task. This section briefly introduces technologies involved in the research process, including the BERT language model used in text preprocessing, CNN and BiGRU networks used in feature extraction, and the attention mechanism.

Text Preprocessing

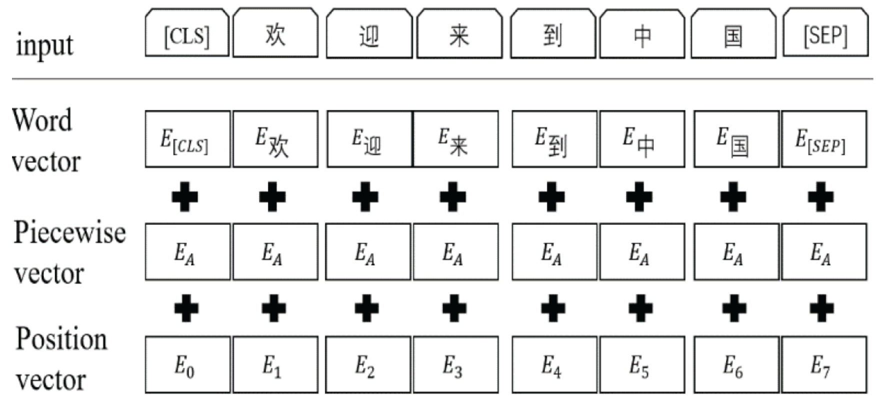
In 2018, a language that is based on the transformer model WAS put forward, i.e., BERT (bidirectional encoder representations from transformers). BERT is a language model that can be used to solve NLP problems. It is widely used in natural language processing tasks such as spam filtering, sentiment analysis, and question-and-answer systems. In traditional text preprocessing tools, such as Word2Vec and Glove (Pennington et al., 2014), the generated text word vectors were static, and such vectors corresponding to the same word in different contexts were unique. Therefore, they could not reflect the semantic information of the text. BERT uses a more efficient two-way transformer encoder than the traditional left-to-right or right-to-left training of text, thus generating dynamic word vectors that can be context-sensitive and adjusted according to different contexts. The input of the BERT model is a combination of three vectors: character embedding representing the current character encoding, the piecewise vector representing the sentence encoding of the current character, and positional coding representing the encoding of the current character. To prepare the vectors, we need to add the tags *[CLS]* and *[SEP]* at the beginning and end of each sentence, respectively. By introducing the position vector, the position of each character can be determined, representing the distance between different characters in the sequence. The input to the BERT model is shown in Figure 1.

Convolutional Neural Network

A convolutional neural network (Albawi et al., 2017) is a deep feedforward neural network based on convolutional computation, which usually contains an input layer, one or more groups of convolutional layers, a pooling layer, a full connection layer, and an output layer. Since a convolutional neural network can automatically learn data features from data sets, it has become one of the most popular deep learning algorithms.

The traditional neural network model generally uses a multi-layer perceptron, which means a fully connected network, that is, each neuron is connected to all the neurons in the next layer. This kind of network requires a large number of training parameters and is prone to the overfitting phenomenon. Regularization is a way to reduce risk by adding additional information to prevent overfitting of the model. Although the model can show better performance by modifying weights and other methods in the process of training the model, such optimization may lead to overfitting. Therefore, a penalty term is added to reduce the value of the global loss function and prevent the model from overfitting.

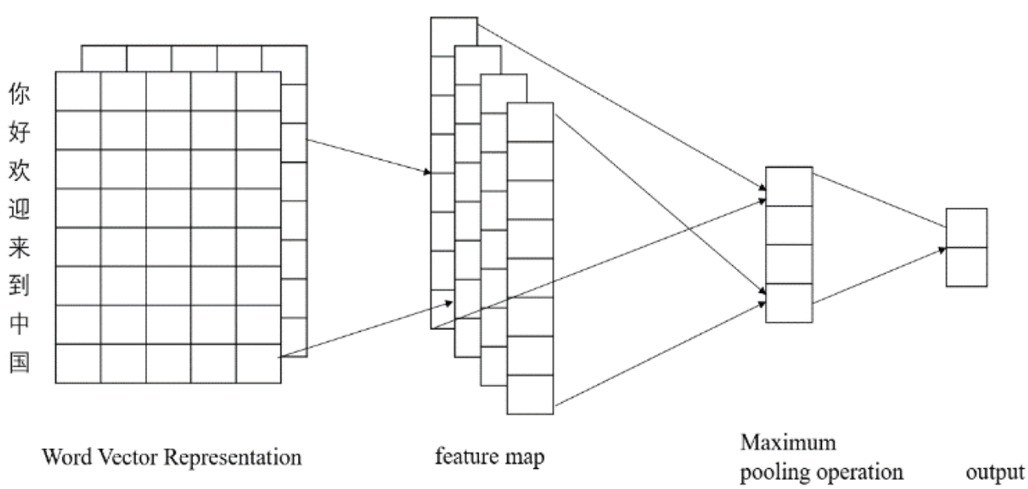
Figure 1. Input of BERT model



Convolutional neural networks can be regarded as regularization of multi-layer perceptron using the idea of hierarchical patterns. The smaller patterns of each layer are combined to form a more complex pattern, which is used as the unit for feature selection. The CNN's structure is inspired by biological visual processes, in which small neurons can see only a limited area, but a combination of many neurons can cover the entire field of vision. A CNN uses less preprocessing, so the model has to learn the filter to be used by itself during training, instead of manually setting the parameters of the filter as in traditional machine learning methods. The advantage of the CNN is that it does not rely on prior knowledge but independently recognizes features (Li et al., 2023). The model structure of a convolutional neural network is shown in Figure 2.

The convolutional layer takes advantage of the spatial locality of data to extract local features of text or images (O'Shea et al., 2015). In essence, the convolutional layer uses convolution cores of different sizes to carry out convolution operations on input data through dynamic training of parameters and capture multiple different features of text or images. In the process of convolution, we may set the step size, which means that the filter can select different spans when scanning. The

Figure 2. Schematic diagram of CNN model structure



convolutional layer adopts the technology of local perception. Compared with the traditional fully connected layer where each neuron is connected to all the neurons of the previous layer, the neurons of the convolutional layer are only connected to specific neurons, reducing the amount of computation and the risk of overfitting. The pooling layer is a down-sampling operation. It is used to reduce the feature space of the feature map. The pooling method used in this paper is maximum pooling, that is, reserving the maximum value in the specified region of the vector output by the filter.

After the convolutional layer and pooling layer, the relevant data features are extracted and passed into the full connection layer and the output layer, and the softmax function is used to output the probability of classification.

Gated Recurrent Unit

The output of the current moment in the RNN is determined by the output of the previous moment and the current input, so the RNN can better learn the representation of the data with long distance dependence. Although the traditional cyclic neural network, e.g., RNN, can capture the dependence information between sequences, it is prone to gradient disappearance and gradient explosion. Moreover, as the length of the input sequence increases, the capability of capturing long distance dependence is poor. LSTM is a variant form of the RNN that uses gate structures to store long distance sequence contents. LSTM has multiple gated units, so it requires a large number of parameters and a long training time. A GRU is a variant of LSTM with good effect. Compared with LSTM, The GRU model structure is simpler. Three gates of LSTM are simplified into two gates, and information saved or abandoned can be selected through the opening of doors (Sherstinsky et al., 2020).

GRUs are simplified on the basis of LSTM, leaving only two gate control units: reset gate and update gate. The update gate is used to control how much of the information that was passed in the previous moment needs to be passed on. The larger the value of the update gate is, the more information from the previous moment is introduced into the current moment. It determines how much information from the previous moment should be passed on. The reset gate determines how well the current input merges with the previous state. The design of GRU update and reset gates can help models learn how to save and forget information from previous moments, thus better capturing long-distance dependencies and effectively alleviating the problems of gradient vanishing and exploding.

The reset gate, which has a value between 0 and 1, is used to control the degree of previous memory introduction in order to calculate the candidate state of the current moment in combination with the current input. If the value of the reset gate is 0, the previously passed information will be completely discarded. GRUs use update gates to complete the operation of status updates and control the inflow of information in the form of gating. It determines the candidate state of the current moment and what information needs to be collected in the memory transmitted from the previous moment.

The gating unit will not automatically clear the information of the previous moment with the flow of time but will decide the information to be retained and forgotten according to the gating situation of the reset and update gates of the current time step and then synthesize the vector to be transferred to the next cell. Therefore, the GRU actively selects the required information, effectively avoiding the problem of gradient disappearance. At the same time, the information is filtered through the gated unit, which reduces the computational complexity and can remember and store the information with a large time interval, and solves the problem that traditional RNNs cannot effectively process the long sequence of text.

Although GRUs can solve the dependency problem between long sequences, they can only receive the information transmitted at the previous time (Liu et al., 2019). Because the context of the text is contextual, the sequence needs to be modeled from both the front and the back. This paper uses the BiGRU model, which consists of a front-to-back GRU and a back-to-front GRU. The output of the BiGRU at each moment is the concatenation of the output vectors of the GRUs in these two directions.

Attention Mechanism

The mechanism of attention stems from the human visual system. When we look at things in the outside world, we don't look at them as a whole, but selectively focus on the more important parts first. The Attention mechanism is to assign different weights to the semantic contribution of input data, extract more critical and important information, and make the model more accurate judgment, but it will not bring more overhead to the model, thus it is widely used. When processing text sequences, each word in the text has a different contribution to the emotional attributes of the text. Introducing the attention mechanism can better extract key information from the text and attach greater weight to words with richer emotional connotations.

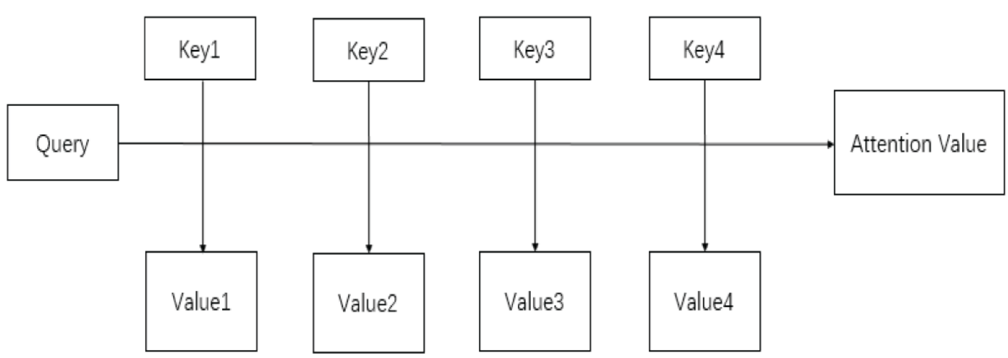
The attention mechanism computes based on text vectors at the source and target sides, and the result is the dependency between each character at the source and each character at the target. In text processing, the self-attention mechanism is selected. It is an attention mechanism only related to the input itself of the source end or the target end. It can also be considered as an attention mechanism adopted when the source end and the target end are the same, capturing the dependency between each word of the source end itself (Niu et al., 2021). Three matrices, denoted as query, key, and value, represent query, key, and value. The attention function can be thought of as an output of the mapping between a query and a key-value pair. The output is a weighted sum of the values calculated by the functions query and value.

When calculating the attention score of the text, query, key, and value are the same, which are obtained by multiplying the text word vector and the three weight matrices, respectively. Thus, when a function is used to calculate the attention score, what is being calculated is the association of the current word with other words in the text sequence. When calculating, the model combines the representation of the word/vector that has been processed previously with the representation of the word/vector that is being processed currently, hence the self-attention mechanism incorporates the understanding of all relevant words into the word vector that we are processing.

The self-attention mechanism can capture semantic or sentential features between different characters in the same text sequence and can capture the long distance dependencies in the text sequence. Before the introduction of the attention mechanism, we did not focus on the position of each word in the text sequence or its specific meaning. After the introduction of attention, different words can be assigned different weights according to the dependencies between the words within the sequence to capture deeper semantic features.

When calculating the attention score, you need to do the query and key to get the corresponding weight. A softmax function is then used to gauge the weight of each value. The resulting score is positive and the sum is 1. This softmax score determines the contribution of each word in encoding the current word, i.e., the weight corresponding to each value. The value vector is then summed

Figure 3. The nature of the attention mechanism



according to the softmax score. The result of self-attention layer at this position is obtained. The self-attention mechanism is more concerned with characters that are more semantically related. The calculation process of the self-attention mechanism can also be understood as the weighted sum of all word representations according to their importance when encoding a word, and the weight determined by the dependency between each word and the current word.

In the actual operation, we compute the Attention function of the query set simultaneously key, query, and value are processed into matrices K, Q, and V, respectively, where d_k represents the dimensions of the query. The calculation method of attention score is shown in Equation 1:

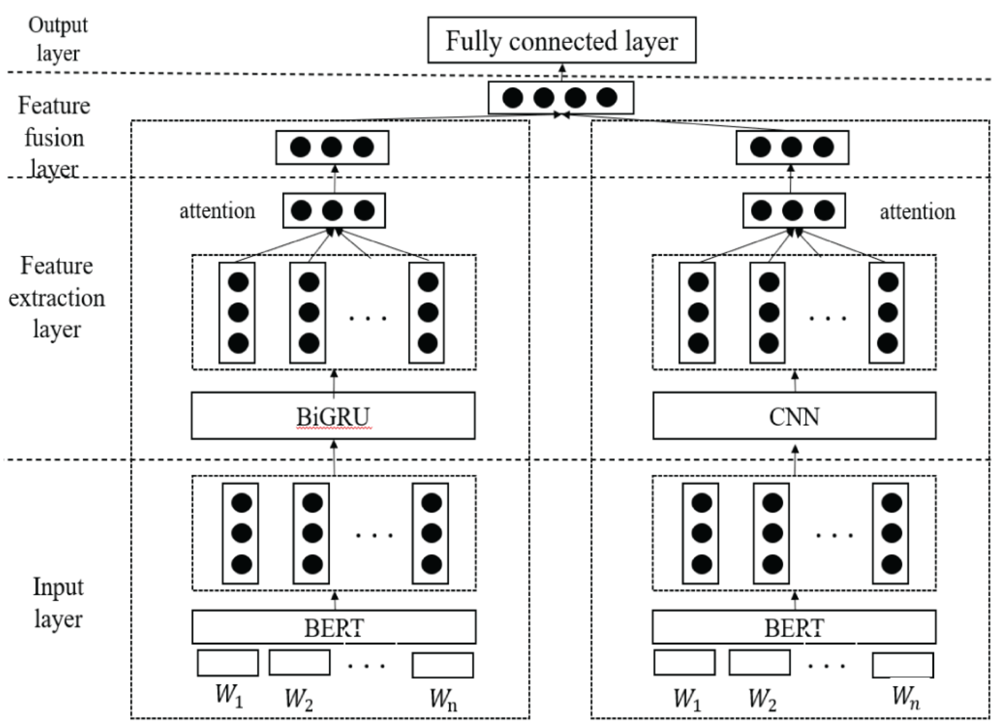
$$Attention\left(Q,K,V\right) = softmax\left(\frac{QK^T}{\sqrt{\left(d_k\right)}}\right) \tag{1}$$

HYBRID NETWORK SENTIMENT ANALYSIS MODEL BASED ON ATTENTION MECHANISM

This paper proposes a hybrid network sentiment analysis model based on attention mechanism. The model structure is shown in Figure 4.

The model is divided into four layers: the first layer is the input layer, in which the text sequence is transformed into the text word vector matrix through BERT language model for downstream tasks. The second layer is the feature extraction layer. Text vectors are input into a CNN and BiGRU,

Figure 4. Mixed network sentiment analysis model based on attention mechanism



respectively, to extract text features. The weighted representation of text vectors is obtained through the attention layer. The third layer is the feature fusion layer. The feature vector output of the CNN and BiGRU is spliced to obtain the global feature vector. The fourth layer is the sentiment analysis layer, which uses a full-connection layer to process the feature vectors and determines the sentiment polarity of the text by output the classification probability of the text through softmax function.

Input Layer

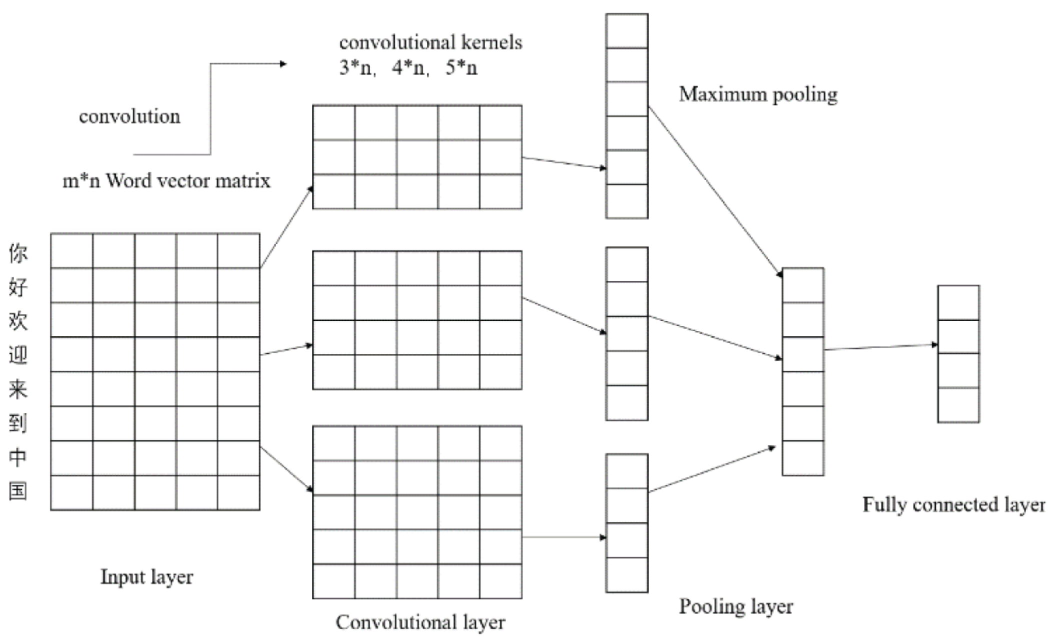
Firstly, the BERT language model is used to pre-train text, and the word vector corresponding to the text is obtained and used as the input of the downstream task for the next step of sentiment analysis. Since BERT uses characters as the unit for training, it is unnecessary to carry out word segmentation and other operations on the text, which can be directly input into the BERT model. The BERT word vector is based on a bidirectional transformer structure and adopts the mask language model and next sentence prediction strategy to train data to integrate the relationship between the context information and sentences and obtain bidirectional semantic information. Each comment text can be represented as a word vector matrix $W = v_1 \oplus v_2 \oplus ... \oplus v_l$, where $\oplus$ represents the connection operation between vectors, and v_i represents the word vector corresponding to each character in the text.

Feature Extraction Layer

Local Feature Extraction

In the feature extraction layer, both CNN and Bi-GRU neural network models are used for feature extraction of word vectors. The convolution kernel is used to recognize text features in the convolution layer. There is still a large amount of redundancy in the output obtained by the convolutional layer. Hence, the pooling layer is introduced to reduce and process the redundant information, further compressing the features and reducing the number of parameters. The model structure of the convolutional neural network used in this paper to extract text features is shown in Figure 5.

Figure 5. Mixed network sentiment analysis model based on an attention mechanism



The parameter received in the input layer of the CNN is the word vector matrix of the text obtained by the BERT model, and the size limit is $m \times n$. Padding is used to fill the text sequence if the number of characters is less than m, and truncate the text sequence if the number exceeds m. The core of the convolution layer is a convolution kernel, and the convolution operation is carried out on the vector matrix. In this paper, three kinds of convolution cores are selected, with sizes of 3, 4, and 5 respectively. For the convolution kernel with window size r, the local feature c_j extracted by the JTH convolution operation is shown as follows:

$$c_j = f\left(w \cdot v_{j:j+r-1} + b\right) \tag{2}$$

where f is the activation function; w is the parameter in the convolution kernel; b is the offset item; $v_j : j + r - 1$ indicates that the convolution kernel reads the word vector of row r once from row j to row $j + r - 1$ of the text matrix. The convolution kernel slides from top to bottom in the text matrix for a total of $n - r + 1$ convolution operations, and the extracted local feature matrix C is represented as follows:

$$C = \left[c_1, c_2, \ldots, c_{n-r+1}\right] \tag{3}$$

The pooling layer is used for further feature extraction. Here, the max pooling method is used, denoted as:

$$\dot{C} = \max\left\{C\right\} \tag{4}$$

The maximum value of the vector is obtained by the convolution layer, and finally the feature vector of a specific length through the fully connected layer is the output.

Global Feature Extraction

Although a CNN can effectively extract local features, it is poor in processing long sequences. Thus, a BiGRU is introduced to supplement training. A BiGRU consists of a forward propagating GRU and a backward propagating GRU and can better consider the semantic information of the text context for global feature extraction. The internal structure of GRUs is shown in Figure 6.

Among them, z_t means the update gate, r_t indicates the reset gate, h_{t-1} represents the state at time t - 1, x_t represents the input at time t, $\tilde{h}_t$ represents the output of x_t and h_{t-1}. The candidate hidden state at time t is obtained. r_t through joint calculation. The value of t determines the degree of forgetting about the hidden state of the previous moment, z_t decides to use $\tilde{h}_t$ and h_{t-1} to update the level of hidden layer state at the current time. The operation process of the GRU is shown in Equations 5–8:

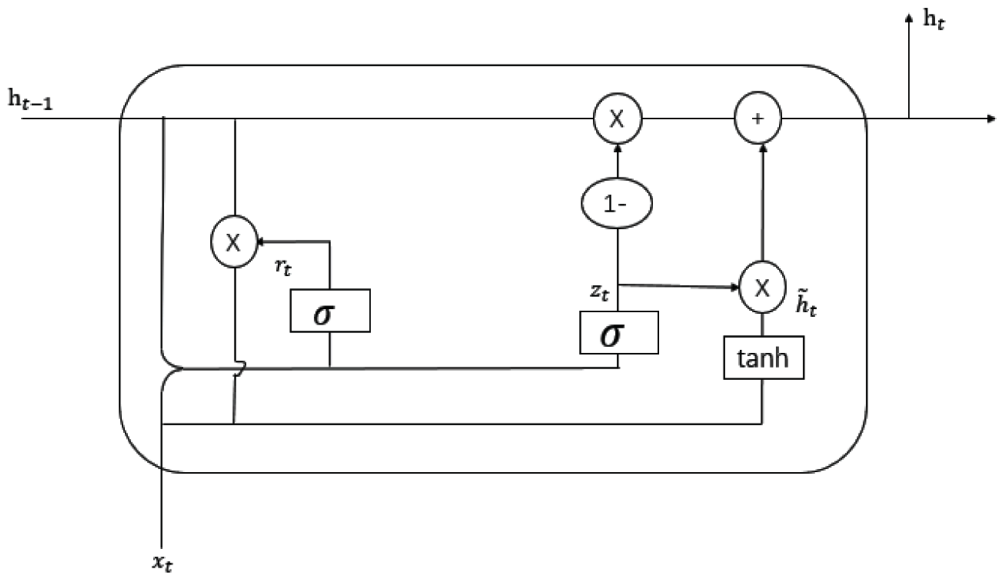
$$z_t = \sigma\left(W_Z \cdot \left[h_{t-1}, x_t\right]\right) \tag{5}$$

$$r_t = \sigma\left(W_r \cdot \left[h_{t-1}, x_t\right]\right) \tag{6}$$

$$\tilde{h}_t = \tanh(W \cdot \left[h_{t-1} * r_t, x_t\right] \tag{7}$$

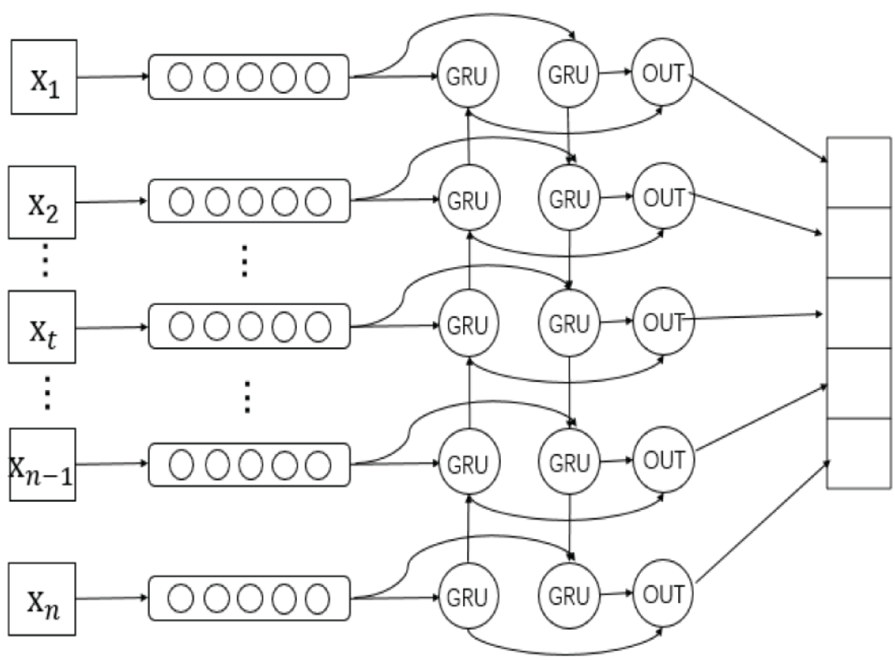
$$h_t = \left(1 - z_t\right) * h_{t-1} + z_t * \tilde{h}_t \tag{8}$$

Figure 6. Schematic diagram of GRU internal structure



At each step, the reset and update gates of the GRU will selectively pass information according to the current input and the hidden state of the previous step so that the previous information can be retained while incorporating the current input. The model structure of BiGRU is shown in Figure 7.

Figure 7. Structure diagram of BiGRU model



GRU helped the input for the current time *t* word vector input x_t and previously hidden layer outputs h_{t-1}, the output of the current moment to remember h_t, *W* and *U* models for training weight matrices. Since a single GRUs can only extract text features from a single direction, bidirectional GRUs are used to extract contextual semantic information from text at the same time. The process of BiGRU calculating output s at time t is shown in Equations 9–12:

$$h_t = Wx_t + Uh_{t-1} \tag{9}$$

$$\vec{h} = \overrightarrow{GRU}h_t \tag{10}$$

$$\overleftarrow{h} = \overrightarrow{GRU}h_t \tag{11}$$

$$s = \left[\overleftarrow{h}, \vec{h}\right] \tag{12}$$

Feature Fusion Layer

Two neural network models, CNN and Bi-GRU, extract the feature of the text word vector matrix, respectively, to obtain two feature vectors, which are spliced together to carry out feature fusion to obtain the global feature vector. In order to simplify the calculation, the feature concatenation here adopts the way of line concatenation.

Sentiment Analysis Layer

The global feature vectors obtained by the splicing of the feature fusion layer are judged through the full connection layer, and the final classification results are output. The model is trained constantly to obtain the weight matrix corresponding to the full connection layer. By multiplying it with the input vector and adding the corresponding bias, the classification result can be obtained. Since it is a binary task, the softmax function is used to map the feature vector to two real numbers between 0 and 1 that sum to 1 as the probability of the text category predicted by the model.

EXPERIMENTAL RESULTS AND ANALYSIS

Experimental Setting

Data Set Selection

In order to verify the performance of the attention-mechanism-based hybrid neural network sentiment analysis model designed in this paper, two Chinese data sets, namely ChnSentiCorp hotel review data set and NLPcc2013-NLPcc2014 microblog review data set, were used for experiments. ChnSentiCorp is a small balanced data set with a longer text length. Microblog comment data set is a large and balanced data set with short text length.

The Hotel Reviews Dataset is a dataset compiled by Tan Songbo and can be downloaded from the CSDN blog. The Weibo comment data set comes from the data set published in NLPcc2013 and NLPcc2014 competitions. The two data sets are downloaded and merged to obtain a larger data set.

Table 1. Data sets

Data set	Categories	Total	Positive	Negative
Hotel review data set	2	4000	2000	2000
Microblog comment data set	2	10000	5500	4500

Evaluation Index

The model in this paper conducts sentiment analysis on text binary classification. In binary classification problems, classification results are usually divided into four categories according to the difference between the actual classification category and the model prediction category: There are four classes of TP (true positive), TN (true negative), FN (false negative) and FP (false positive). The confusion matrix of these four categories is shown in Table 2.

In the data set selected in this paper, *TP* represents the number of samples correctly predicted by the model in the text with positive actual emotional attribute. *TN* represents the number of samples correctly predicted by the model in the text with negative actual emotional polarity. *FN* represents the number of samples predicted by the model in the text with positive emotional polarity. *FP* represents the number of samples in which the text with actual negative emotional polarity is predicted to be positive.

The above parameters can be used to obtain the measurement index *P* value and *F* value of the model, which represent the accuracy rate and recall rate of the model prediction, respectively. The methods for calculating the accuracy rate and recall rate are shown in Equations 13–14, respectively:

$$precision = \frac{TP}{TP + FP} \tag{13}$$

$$recall = \frac{TP}{TP + FN} \tag{14}$$

The accuracy rate measures the proportion of samples that are actually positive among those whose emotional attributes the model predicts are positive. The recall rate represents the proportion of samples with positive actual emotional attributes predicted correctly by the model. Because the precision ratio and recall ratio sometimes contradict each other, another measurement index is introduced: the F1 value. The calculation formula is shown in Equation 15:

$$F1 = \frac{2PR}{P + R} \tag{15}$$

where P is the precision ratio and R is the recall ratio.

The most intuitive index to describe the performance of the classification model is the accuracy rate (Acc) of the model, that is, the proportion of correctly classified samples in all samples. Generally speaking, the model with higher accuracy rate has better performance. The calculation formula of accuracy rate is shown in Formula (16):

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{16}$$

Table 2. Confusion matrix

	True positive sample	**True negative sample**
Predict positive samples	*TP*	*FP*
Predict negative samples	*FN*	*TN*

In addition, this paper also introduces the ROC-AUC index, which is often used to measure the model performance in binary tasks. The evaluation criteria are as follows: in the ROC curve drawn, the closer the ROC curve is to the upper left corner, the better the performance of the model is. The larger the value of AUC, the better the performance of the model.

Experimental Reference System and Related Parameter Settings

In order to verify the effectiveness of the hybrid neural network sentiment analysis model based on the attention mechanism designed in this paper, five models were set up as baseline models for comparative experiments. The baseline models include CNN-fully connected-attention, CNN-BiGRU, CNN-BiLSTM-Attention, CNN-attention, and BiGRU-attention.

The parameter settings of different models are shown in Table 3.

Experimental Results and Analysis

In order to verify the performance of the hybrid neural network sentiment analysis model based on attention mechanism proposed in this paper, it was compared with several baseline models in two datasets: the hotel review dataset and the Weibo review dataset. The accuracy rate, F1 value, and AUC value displayed by different models in the hotel review data set and microblog review data set are shown in Table 4.

As can be seen from Table 5, the hybrid neural network sentiment analysis model based on the attention mechanism proposed in this paper performs well and achieves good results in both the hotel review and microblog review data sets. Compared with the model composed of a single neural network, such as CNN Attention, BiGRU-attention and CNN-fully connected-attention, the accuracy

Table 3. Experimental parameter settings

Parameter name	Parameter meaning	Parameter setting
epoch	Number of sample iterations	Hotel review:10/microblog:5
batch_size	Batch sample size	64
learning_rate	Learning rate	0.001
dropout	The proportion of neurons that are discarded	0.5
optimization	Optimization algorithm	Adam
cell	The number of neurons in LSTM and GRU	64
kernel_size	CNN convolution kernel size	3, 4, 5

Table 4. Classification accuracy, F1 value, and AUC value of the model

Data set	Evaluation index	Hybrid network model based on attention mechanism	CNN-fully connected-attention	CNN-BiGRU	CNN-BiLSTM-attention	CNN-attention	BiGRU-attention
Hotel review data set	Acc	0.919	0.916	0.908	0.9	0.918	0.873
	F1	0.92	0.92	0.914	0.904	0.92	0.879
	AUC	0.97	0.965	0.966	0.968	0.969	0.923
Microblog comment data set	Acc	0.875	0.862	0.872	0.859	0.874	0.816
	F1	0.870	0.861	0.870	0.868	0.881	0.814
	AUC	0.936	0.927	0.922	0.928	0.934	0.834

of the model proposed in this paper increased by 0.1%, 4.3% and 0.2%, respectively, on the hotel review data set. The accuracy of the test on microblog comment data set increased by 0.6%, 4.9% and 1.3%, respectively. Thus, it can be seen that a single neural network may have some problems in the construction of the model when extracting text features, while a hybrid neural network can complement each other's advantages. CNNs can extract the local features of text well, but it cannot capture the long-distance dependence of text sequences. BiGRUs can save historical information and effectively make up for the shortcomings of CNNs, thus the hybrid neural network performs better. At the same time, compared with the CNN-BiLSTM-attention benchmark model, the accuracy of this model in hotel review and Weibo review data sets increased by 1.8% and 1.6%, respectively. BiGRU and BiLSTM are both variant forms of cyclic neural networks. GRUs and LSTM both introduce the concept of gates, but a GRU is further simplified based on LSTM, reducing the number of training parameters, and effectively avoiding the situation of gradient vanishing and gradient explosion. Compared with the CNN-BiGRU benchmark model without the introduction of the attention mechanism, the accuracy of the proposed model in the two data sets is improved by 1% and 0.3%, respectively, indicating that the attention mechanism can better capture features and improve the classification accuracy.

CONCLUSION

With the rapid development of the Internet, the amount of data increases geometrically. How to analyze data to get valuable content is a hot topic at present. It is important to analyze the content with emotional attributes on the Internet and dig out its intrinsic information. In this paper, based on research outside the field of sentiment analysis, a hybrid neural network sentiment analysis model based on the attention mechanism is proposed. The attention mechanism is integrated into the convolutional neural network and the hybrid neural network of bidirectional gated circulation units, and sentiment analysis is carried out on the text. The relevant experiments are carried out on the hotel review and microblog review datasets. The model performance was evaluated and compared. The experimental results show that compared with the baseline model, the mixed neural network sentiment analysis model proposed in this paper has improved the classification accuracy and F1 value, further improving the accuracy of sentiment analysis of text, and the model performs better. Compared with the single neural network model, the mixed neural network sentiment analysis model designed in this paper has improved the accuracy of sentiment classification, but it still has some shortcomings such as too long training time and too much computation. Deep learning has been applied more and more deeply and widely in the field of sentiment analysis, and the overall performance of the model needs to be further improved. There are still some problems worth further research:

- How to integrate the text characteristics studied in linguistics into our training model, so that the model can be trained jointly from both parameters and actual contexts, so as to obtain a model with better performance.
- A statement may express different emotions for different entities. How to classify emotions at a finer granularity in the text.

ACKNOWLEDGMENT

This work is partly supported by the Fundamental Research Funds for the Central Universities.

REFERENCES

Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. *Proceedings of the 2017 International Conference on Engineering and Technology*, (pp. 1–6). IEEE.

Bahdanau, D., Cho, K., & Bengio, Y. (2015). *Neural machine translation by jointly learning to align and translate*. CoRR.

Cho, K., van Merrienboerb, Gulcehrec, et al. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language*, (pp. 1724–1734). ACL. doi:10.3115/v1/D14-1179

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of deep bidirectional transformers for language understanding*. NAACL-HLT.

Hemmatian, F., & Sohrabi, M. K. (2019). A survey on classification techniques for opinion mining and sentiment analysis. *Artificial Intelligence Review*, *52*(3), 1495–1545. doi:10.1007/s10462-017-9599-6

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. doi:10.1162/neco.1997.9.8.1735 PMID:9377276

Kalchbrenner, N., Grefenstette, E., & Blunsom, P. (2014). *A convolutional neural network for modelling sentences*. arXiv preprint arXiv:1404.2188. 10.3115/v1/P14-1062

Kim, Y. (2014). *Convolutional neural networks for sentence classification*. EMNLP. doi:10.3115/v1/D14-1181

Li, W., Guo, C., Liu, Y., Zhou, X., Jin, Q., & Xin, M. (2023). Rumor source localization in social networks based on infection potential energy. *Information Sciences*, *634*, 172–188. doi:10.1016/j.ins.2023.03.098

Liu, G., & Guo, J. (2019). Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing*, *337*, 325–338. doi:10.1016/j.neucom.2019.01.078

Liu, J., Yang, Y., & Lv, S. (2019). Attention-based BiGRU-CNN for Chinese question classification. *Journal of Ambient Intelligence and Humanized Computing*, 1–12.

Mikolov, T., Sutskever, I., & Chen, K. (2013). Distributed representations of words and phrases and their compositionality. *Proceedings of the 26th International Conference on Neural Information Processing Systems, 26*, (pp. 3111–3119). IEEE.

Niu, Z., Zhong, G., & Yu, H. (2021). A review on the attention mechanism of deep learning. *Neurocomputing*, *452*, 48–62. doi:10.1016/j.neucom.2021.03.091

O'Shea, K., & Nash, R. (2015). *An introduction to convolutional neural networks*. arXiv preprint arXiv:1511.08458.

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, *2*(1/2), 1–35. doi:10.1561/1500000011

Pennington, J., Socher, R., & Manning, C. (2014). GloVe: Global vectors for word representation. *Proceedings of 2014 Conference on Empirical Methods in Natura Language Processing*, (pp. 1532–1543). ACL. doi:10.3115/v1/D14-1162

Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D. Nonlinear Phenomena*, *404*, 132306. doi:10.1016/j.physd.2019.132306

Yang, Z. C., Yang, D. Y., & Dyer, C. (2016). Hierarchical attention networks for document classification. *Proceedings of 2016 Conference of the North American Chapter of the Association for Computational Linguistics*, (pp. 1480–1489). ACL.

Zhang, S. X., Wei, Z. L., Wang, Y., & Liao, T. (2018). Sentiment analysis of Chinese micro-blog text based on extended sentiment dictionary. *Future Generation Computer Systems*, *81*, 395–403. doi:10.1016/j.future.2017.09.048

Zhu, X., Sobhani, P., & Guo, H. (2015). *Long short-term memory over recursive structures*. ICML.

Wenqian Shang received bachelor's, master's, and PhD degrees from Southeast University in Nanjing, China, in 1994, National University of Defense Technology, Changsha, China, in 1999, and Beijing Jiaotong University, Beijing, China, in 2008, respectively. She finished her postdoctoral work at Communication University of China in 2013. Her current appointment is a professor at the Communication University of China. Professor Shang was the program chair of IEEE/ACIS ICCIS 2019 and conference chair of CSII 2021. She has published more than 60 journal articles and conference papers, in such venues as Expert Systems with Applications and the Journal of Computer Research and Development. Her current research interests include artificial intelligence, machine learning, and NLP.

Hongzhan Zhen Graduated in 2022 from Zhengzhou University of Light Industry with a Bachelor's degree in software engineering. Currently pursuing a Master's degree in Big Data Technology at Communication University of China. His current research interests include false news detection and sentiment analysis.

Wanyu Zhang has received a bachelor's degree from Communication University of China in Beijing, China, in 2021. She is applying for a master's degree in computer science and technology at Communication University of China. Her current research interests include federated learning and machine unlearning.