Intelligent Automation & Soft Computing

Tech Science Press

DOI: 10.32604/iasc.2023.037900
Article

A PERT-BiLSTM-Att Model for Online Public Opinion Text Sentiment Analysis

Mingyong Li, Zheng Jiang*, Zongwei Zhao and Longfei Ma

College of Computer and Information Science, Chongqing Normal University, Chongqing, 401331, China
*Corresponding Author: Zheng Jiang. Email: jiangzheng941@126.com
Received: 20 November 2022; Accepted: 10 March 2023; Published: 23 June 2023

Abstract: As an essential category of public event management and control, sentiment analysis of online public opinion text plays a vital role in public opinion early warning, network rumor management, and netizens' personality portraits under massive public opinion data. The traditional sentiment analysis model is not sensitive to the location information of words, it is difficult to solve the problem of polysemy, and the learning representation ability of long and short sentences is very different, which leads to the low accuracy of sentiment classification. This paper proposes a sentiment analysis model PERT-BiLSTM-Att for public opinion text based on the pre-training model of the disordered language model, bidirectional long-term and short-term memory network and attention mechanism. The model first uses the PERT model pre-trained from the lexical location information of a large amount of corpus to process the text data and obtain the dynamic feature representation of the text. Then the semantic features are input into BiLSTM to learn context sequence information and enhance the model's ability to represent long sequences. Finally, the attention mechanism is used to focus on the words that contribute more to the overall emotional tendency to make up for the lack of short text representation ability of the traditional model, and then the classification results are output through the fully connected network. The experimental results show that the classification accuracy of the model on NLPCC14 and weibo_senti_100k public data sets reach 88.56% and 97.05%, respectively, and the accuracy reaches 95.95% on the data set MDC22 composed of Meituan, Dianping and Ctrip comment. It proves that the model has a good effect on sentiment analysis of online public opinion texts on different platforms. The experimental results on different datasets verify the model's effectiveness in applying sentiment analysis of texts. At the same time, the model has a strong generalization ability and can achieve good results for sentiment analysis of datasets in different fields.

Keywords: Natural language processing; PERT; pre-training model; emotional analysis; BiLSTM

1 Introduction

With the rapid development of social platforms such as Weibo, Facebook and Twitter, and the increasing number of users of social network service platforms such as forums, many public opinion text data with vibrant color have been generated. The comment text of social platforms belongs to the category of short text. The information is short, lacks annotation data, and has the characteristics of colloquialism and many new words on the network [1]. Online public opinion is the mapping of public social opinion in the Internet space and is the direct reflection of public social opinion [2]. The outbreak of public opinion events often has sudden asymmetry, and a large number of comments (significantly negative comments) in a short time often mark the outbreak of a public opinion event. Therefore, to realize the monitoring and management of network public opinion and prevent a sudden public opinion crisis, the efficient sentiment analysis method of public opinion text is of great significance to the early warning, practical guidance and benign development of public opinion.

For comment texts on social platforms, the position information of words plays a vital role in the overall semantics, likely, the change of the position of the headword in a sentence will lead to a change in the overall semantics. In addition, in daily reading, we may not notice some marginal words in the sentence, but it is still easy to grasp the central meaning of the sentence through the keywords. It is not enough to grasp the keywords to understand the sentence's meaning. For example, "My new phone broke the first day I had it. It's a surprise!". The traditional sentiment analysis model will be sensitive to the keyword "surprise" and lead to a positive prediction of the sentence's meaning. This shows that the existing sentiment analysis model only focuses on the keywords in the sentence and is not sensitive to the sentence's word order. Pre-trained models commonly used in natural language processing, such as Bidirectional Encoder Representation from Transformers (BERT) [3], are prone to these problems. It is challenging to learn the position information of the word while focusing on the keywords, which makes it difficult to achieve the expected accuracy of sentiment tendency prediction, especially for short texts such as social platforms. In view of the above problems, this paper proposes a public opinion text sentiment analysis model, which combines the Pre-training BERT with Permuted Language Model (PERT) [4], Bidirectional Long Short-term Memory Network (BiLSTM) and Attention Mechanism (Att). The main contributions of the PERT-BiLSTM-Att model are as follows:

1. A model combining PERT, BILSTM and Attention mechanism is proposed. As far as we know, this is the first work to combine the out-of-order language model PERT with BiLSTM and attention mechanism for sentiment classification. On the one hand, the model utilizes the PERT model pre-trained from a large amount of word position information to obtain the vector representation of sentences, which enhances the learning of word position information in sentences. On the other hand, using BiLSTM to extract bidirectional features of text enhances the learning ability of time series, which also enhances the model's ability to represent long and complicated sentences. Finally, to enhance the effect of sentiment classification for short texts, we use the attention mechanism to learn the importance of each word in the sentence fully. The whole model considers the context information of the sentence and the influence of some keywords on the emotional tendency of the whole sentence, which enhances the effect of emotional classification.
2. The attention mechanism is introduced to make full use of the characteristics of solid colloquialism and frequent occurrence of emotional words in the network public opinion text so that the characteristics that may affect the classification of emotional polarity in the text are more apparent. The model can also focus on the words with the higher contribution to the text to improve sentiment classification accuracy.

3. The effect of the model is verified on the public data sets NLPCC14 and weibo_senti_100k, and the generalization effect of the model is verified on the MDC22 data set composed of captured Meituan, Dianping and Ctrip.com comments. Ablation experiments also confirm the effectiveness of different modules of the proposed model.

The rest of this article is arranged as follows. The second part introduces the research related to the work of this paper. The third part elaborates on the method we proposed. The experimental results and analysis are shown in the fourth part. Finally, the fifth part summarizes this thesis.

2 Related Work

2.1 Sentiment Analysis

Sentiment analysis is also known as opinion mining in academia. It is a very important sub-field in natural language processing. The research goal of sentiment analysis is to analyze people's emotions, attitudes, and opinions expressed on things in comment texts [5]. Currently, the research methods of sentiment analysis are mainly divided into three categories: sentiment analysis method based on sentiment dictionary, sentiment analysis method based on traditional machine learning and sentiment analysis method based on deep learning.

The method based on the sentiment dictionary needs to construct an emotional dictionary in the relevant fields involved. By finding the keywords of the dictionary and combining the rules of manual design, we can judge the emotional tendency of the text. For example, Yanyan et al. [6] expanded the emotional dictionary by constructing many emotional words, and the performance of Weibo emotion classification was improved by 1.13% compared with the baseline method. Song et al. [7] weighted the raw emotional scores, finally obtained the overall emotional scores of the text and refined the text's emotional data to construct a more efficient emotional dictionary. The most significant disadvantage of the method based on a sentiment dictionary is that it is difficult to build a sufficiently comprehensive sentiment dictionary. The sentiment analysis is only suitable for small fields and has yet to have a good universality.

The sentiment analysis method based on traditional machine learning generally refers to the use of techniques such as support vector machine(SVM) [8] and K-nearest neighbor(KNN) algorithm [9] for supervised learning and then predicting the sentiment of the text. For example, Pang et al. [10] first applied machine learning algorithms to sentiment analysis of movie review data; Zhu and others [11] used the bag-of-words model to generate word vectors for text data and combined them with algorithms such as SVM to achieve an improvement in accuracy. The sentiment analysis method based on traditional machine learning does not need to rely too much on the sentiment lexicon. Still, because the generalization ability of the model trained by the machine learning method is general and the previous training requires a lot of manual labeling, it isn't easy to apply in various complex scenes.

The method based on deep learning does not require artificial feature extraction and has strong semantic expression and generalization ability. In recent years, the neural network structure has achieved remarkable results in text classification. The commonly used neural network structures include convolutional neural network(CNN), recurrent neural network(RNN) [12], long short-term memory network(LSTM) [13], gated recurrent unit(GRU) [14], etc. In the application of text classification, Kim et al. [15] first proposed the application of CNN to text orientation analysis and achieved better results than traditional machine learning methods. However, CNN can only extract local features and has poor capture performance for long-distance dependence, which leads to CNN's poor classification of long and complex sentences. To solve this problem, Mikolov et al. [16] proposed

to apply RNN to text sentiment analysis. The feature of RNN is that each node can use the information of previous nodes. Hence, RNN is better than CNN at capturing long-distance dependencies and is more suitable for modeling long-sequence information.nd complex sentences. However, with the increase of input, the perception ability of RNN to early text input decreases, which is prone to problems such as gradient disappearance or gradient explosion. With the emergence of improved RNN networks such as LSTM and GRU, it is possible to model more compresensive sequence information and even the whole article. For example, Du Yongping proposed a CNN-LSTM combined short text sentiment classification method [17]. The F1 values of positive and negative sentiment recognition on the NLPCC evaluation dataset reached 0.7683 and 0.7724, respectively. Yin et al. [18] proposed a multi-channel LSTM imbalanced emotion classification method. This method uses the undersampling method to obtain multiple sets of the balanced training corpus. Fusing multiple LSTM models for classification obtains higher accuracy than CNN, RNN, and other methods.

2.2 Attention Mechanism and Bert

In recent years, attention mechanism has become one of the research focuses in deep learning. The attention mechanism simulates the process of human observation of things. That is, different attention weights are given to different regions, a more significant weight is given to the parts that need to be focused, and a smaller weight is given to the non-focused parts, which makes the attention mechanism easier to extract the parts that need to be focused in the data than traditional deep learning methods such as RNN and CNN. The first application of the attention mechanism is the glimpse algorithm by Mnih et al. [19] in the study of image classification. Unlike full-image scanning, the algorithm only focuses on some areas of the image each time. It integrates the content of multiple concerns with the recurrent neural network in chronological order to establish the dynamic representation of the image. The algorithm reduces the time complexity and noise interference and has achieved remarkable results in the image classification task. Like image processing, natural language processing models can focus on the task-related parts of the text when processing text and pay less attention to the unimportant parts. According to this idea, Bahdanau et al. [20] applied the attention mechanism to the neural machine translation model. That is, when generating each translation term, let the model find the most relevant part of the original text and the current term and predict accordingly. This is the first application of the attention mechanism in natural language processing. Attention does not depend on downstream tasks and improves the representation of the source text sequence. Inspired by this, Lin et al. [21] proposed to use attention to embed sentences to enhance the semantic representation of sentences. Firstly, the hidden layer sequence of sentences is obtained by bidirectional LSTM. Then the weight matrix between all elements of the sentence is calculated by attention, and then the matrix representation of the sentence is obtained. Unlike the traditional practice of embedding sentences into fixed vectors, this work uses self-attention to embed sentences into matrices. Each row of the matrix reflects the semantic features of the sentence for the corresponding elements. The empirical results show that this method can enrich the semantics of sentences more than the fixed vector method. One of the core problems of text classification is feature selection [22]. Compared with previous methods, the attention mechanism can dynamically assign weights to text features so that the classifier can focus on using feature information. Many works have confirmed the effectiveness of the attention mechanism in classification tasks [23–25].

Self-attention is an improvement of the attention mechanism, which is used to model the dependencies between elements within the source text sequence to enhance the understanding of the source text semantics. The internal attention proposed by Cheng et al. [26] is the ideological enlightenment of self-attention. It is a unique attention mechanism within the text sequence, especially suitable for

long-distance dependence capture. Vaswani et al. [27] first proposed Transformer, a pre-trained model based on self-attention training. BERT is a pre-trained model proposed by DevlinJ et al. of the Google AI team in 2018. It is essentially a bi-directional Transformer model. BERT takes into account both left and right Token context information. The modeling learning ability of long sequences is enhanced. Like Transformer, BERT uses a self-attention mechanism in the coding module simultaneously. The overall structure of BERT is shown in Fig. 1.

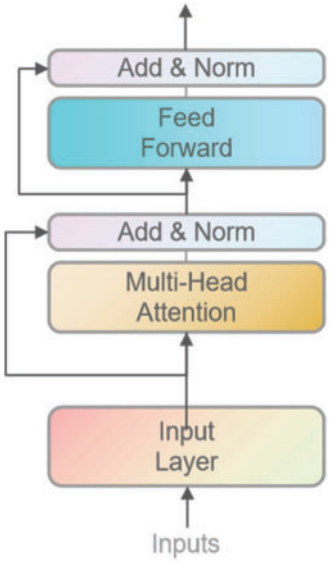


Figure 1: The input and structure of the BERT model

The input layer of the Bert model consists of the sum of three vectors, which are word embedding, sentence embedding, and position embedding [28], as shown in Fig. 2. Word embedding generates a vector representing the current word through WordPiece embedding. Sentence embedding represents the encoding of the sentence where the current word is located, which is used to distinguish two sentences and judge whether the sentence is the context of the corresponding sentence. Position embedding represents the position encoding of the current word so that the model can learn the position information of the word in the sentence. Each sentence uses CLS and SEP as the beginning and end markers.

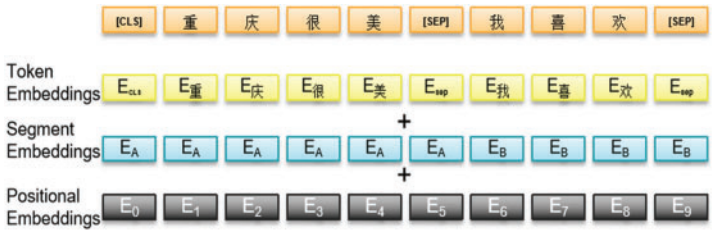


Figure 2: The input of the BERT model

The training of the BERT model is mainly composed of two self-supervised tasks, namely Masked Language Model (MLM) and Next Sentence Prediction (NSP). Among them is Masked Language Model: Through random MASK (masking) 15% of the words in each sentence, the model predicts the

words marked with MASK to train the model's perception of the sentence. Next Sentence Prediction: Allow the model to predict whether a statement pair is relevant by selecting continuous and non-continuous statement pairs from the trained corpus. The input to the Bert model is shown in Fig. 2.

The word vector processed by the input layer enters the Mult-Head-Attention layer composed of h Self-Attention in parallel. After linear transformation and h times of scaling dot product, the attention output is obtained. Next, into the Add Norm layer using Residual Connection [29] and Layer Normalization [30] to avoid problems such as gradient disappearance when the model hierarchy is too deep, the output through the Add Norm layer is then passed to the Feed Forward layer, and finally output after passing through the Add Norm layer.

3 Proposed Method

In this section, we elaborate on our proposed method in detail, including the following sections: pre-training model PERT based on an out-of-order language model, model overview, and model calculation process.

3.1 Pre-Training Model PERT Based on Out-of-Order Language Model

Since the emergence of pre-training models such as Transformer and BERT, according to the different tasks of training models, pre-training models for natural language understanding (NLU) can be divided into two categories by using and not using mask marking [MASK] to mask input text for training directly. Models trained with masked [MASK] input text, represented by BERT, have achieved good results in various downstream tasks of natural language processing, but the fact that some words in BERT are masked increases noise, which reduces the effectiveness of the model to some extent. Moreover, for texts composed of single words, if the central word or keyword of a sentence is masked, it may destroy the emotional tendency of the whole sentence and make the masked words more difficult to predict. For example, 'This store is completely different from what I expected, which is too disappointing.' If the 'disappointment' is masked to allow the model to predict, it may predict various results, such as 'surprise,' 'sad,' etc. This is because the corpus before and after the word 'disappointment' can only judge that the emotional tendency of the sentence is neutral, and for the network public opinion text, the common phenomenon is that the text is short, and it is difficult to judge the emotional polarity through the front and back of the keywords. Often there are only one or two keywords in a sentence. Therefore, improving the pre-training task and the performance of the pre-training model has become a significant research hotspot in recent years.

Studies have shown that a certain degree of disordered text does not affect the overall sentence understanding of text. For example, 'the order of a sentence does not reading affect,' then can we learn semantic knowledge from the disordered text and then train the model? Inspired by the above language features, the Joint Laboratory of HIT and iFLYTEK Research (HFL) proposed the Pre-Training BERT with Permuted Language Model (PERT) in 2022. It achieved good results in multiple NLP tasks. Like BERT, PERT is also an automatic coding model trained by Permuted Language Model (PLM). The training process of PERT is to replace part of the input text. The training goal is to predict the position of the original mark, which is to predict the correct word arrangement of a sentence. PERT uses a large number of Chinese corpus to train. It consists of the Chinese Wikipedia, encyclopedia, community question answering, news articles, etc. The total training data has 5.4B words and takes about 20G disk space.

In addition, the masked words are randomly selected in the BERT model, and PERT uses the Whole Word Masking [31] to replace the random mask during training to improve performance. In

addition, the N-gram mask is used to perform a sliding window operation with a size of N according to bytes, forming a byte fragment sequence with a length of N for segmentation masking to further improve the model's ability to model long sequence learning. The input and structure of the PERT model are shown in Fig. 3, and the comparison of the input and output of the PERT model and the BERT model is shown in Table 1.

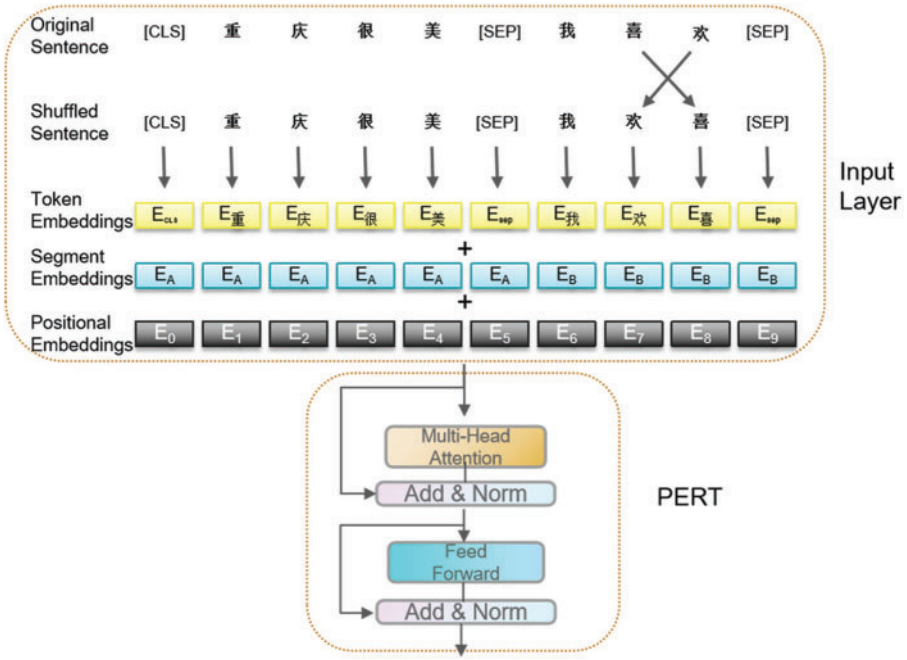


Figure 3: The input and structure of the PERT model

Table 1: The input and output comparison of PERT and BERT

	Input	Output
Original text	Research shows that the order of a sentence does not affect reading.	–
Word piece	Research shows that the order of a sentence does not affect reading.	–
BERT	Research [M] that the order of a sentence does not affect [M].	$Pos_2 \rightarrow$ shows $Pos_{12} \rightarrow$ reading
PERT	Research shows that order the of a sentence does not reading affect.	$Pos_3 \rightarrow Pos_4$ $Pos_4 \rightarrow Pos_3$ $Pos_{11} \rightarrow Pos_{12}$ $Pos_{12} \rightarrow Pos_{11}$

The training process of PERT is as follows. First, given a pair of sentence sequences $A = \{A_1, \ldots, A_n\}$ and $B = \{B_1, \ldots, B_m\}$, Among them, $A_1, \ldots, A_n$ and $B_1, \ldots, B_m$ can be understood as the word vector of the words that make up the sentence sequence. An N-gram masking strategy is used to select candidate labels for the above sentence sequences. The percentage of word-level unigrams

to 4-gram is 40%, 30%, 20%, and 10%, respectively. Similar to BERT, only 15% of the input layer is selected for position masking.

The masked sentence sequences are $A' = \{A'_1, \ldots, A'_n\}$ and $B' = \{B'_1, \ldots, B'_m\}$, and the order of some words has been disrupted. Then we connect the two sequences to form the input layer sequence X of PERT, where [CLS] and [SEP] represent the beginning of the sentence sequence and the separation mark between the two sentences, respectively.

$$X = [\text{CLS}]\, A'_1, \ldots, A'_n\, [\text{SEP}]\, B'_1, \ldots, B'_m\, [\text{SEP}] \tag{1}$$

Then, the PERT model converts the sequence X into the context representation $H \in^{N \times d}$ through the embedding layer, which is composed of word embedding, position embedding, and label type (segment) embedding. Then the embedding layer is input to the Transformer layer of the L layer and processed by the multi-head self-attention mechanism, where N is the maximum sequence length and d is the dimension of the hidden layer. The process that the sentence sequence is processed by the embedding layer and then sent to the multi-head self-attention layer can be represented as follows.

$$H^0 = \text{Embedding}\,(X) \tag{2}$$

$$H^i = \text{Transformer}\left(H^{i-1}\right), i \in \{1, \ldots, L\}, H = H^L \tag{3}$$

Similar to the MLM task in the BERT pre-training task, the PERT model's training only needs to predict the location of the correct word, so it needs to collect a subset of all possible locations to form a candidate representation $H^m \in \mathbb{R}^{k \times d}$, where is the number of markers k selects. Because of the 15% masking rate, $k = [N \times 15\%]$. Next, the feed-forward dense layer (FFN), dropout, and layer normalization layer are used for further processing.

$$\widetilde{H} = \text{LayerNorm}\,(\text{Dropout}\,(\text{FFN}\,(H^m))) \tag{4}$$

To calculate the position of the original marker, a dot product needs to be made between H^m and H to calculate the weight of the correct position to be predicted. To make the model converge better, a bias term $b \in \mathbb{R}^L$ is added, and finally, the probability of normalization is obtained through the SoftMax layer. The following formula, p^i is the probability of the word to be predicted at that position, and the model finally uses the standard cross entropy loss to optimize the pre-training task.

$$p^i = softmax\left(\widetilde{H}_i^m H^T + b\right), p^i \in \mathbb{R}^L \tag{5}$$

$$\gamma = -\frac{1}{M}\sum_{i=1}^{m} y_i \log p^i \tag{6}$$

3.2 Model Overview

The PERT-BiLSTM-Att model structure is shown in Fig. 4. The model is mainly composed of the following parts: input layer, pre-training model processing layer, BiLSTM layer, attention layer, and output layer.

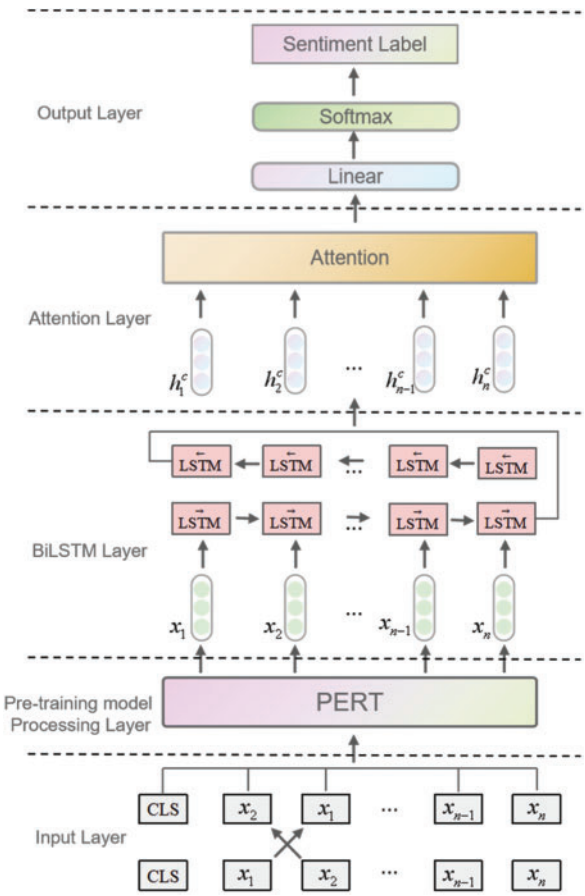


Figure 4: The structure of the PERT-BiLSTM-Att model

3.3 Model Calculation Process

Firstly, the public opinion text data is preprocessed. The text is embedded with position information and sentence information through the input layer to obtain the text sequence $X = (x_1, x_2, \ldots, x_n)$ that is input to the PERT pre-training model processing layer.

The text sequence representation x_t $(1, 2 \ldots, n)$ of the learned position information is obtained by the PERT pre-training model, where x_t represents the word vector representation of the word at the t position.

As a variant of the recurrent neural network RNN, LSTM has better long-sequence semantic capture modeling capabilities than RNN [32]. It can well solve the problems of gradient disappearance or explosion that are easy to occur during the training process of the RNN model through the gating mechanism. The training process of inputting the text feature x_t $(1, 2 \ldots, n)$ obtained by the pre-trained PERT layer into the LSTM network is as follows:

$$f_t = \sigma\left(W_f \cdot [x_t, h_{t-1}] + b_f\right) \tag{7}$$

$$i_i = \sigma\left(W_i \cdot [x_t, h_{t-1}] + b_i\right) \tag{8}$$

$$o_t = \sigma\left(W_o \cdot [x_t, h_{t-1}] + b_o\right) \tag{9}$$

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{10}$$

$$\tilde{c}_t = \tanh(W_c \cdot [x_t, h_{t-1}] + b_c) \tag{11}$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t \tag{12}$$

$$h_t = o_t \cdot \tanh(c_t) \tag{13}$$

In the above formula, x_t is the input of the current moment, and h_{t-1} is the overall state of the previous moment. f_t represents the forgetting gate, which determines the internal state of the previous moment by calculating the current moment which information should be discarded, i_i represents the update gate, which determines how much information the current candidate state $\tilde{c}_t$ should retain, o_t is the output gate, which determines the final overall state output h_t by interacting with the current state c_t. σ is nonlinear activation function and tanh is hyperbolic tangent function. W_f, W_i, W_o and W_c are parameters for calculating the forgetting gate, updating gate, output gate and internal candidate state respectively. b_f, b_i, b_o and b_c correspond to the bias parameters of the above calculation.

Although LSTM solves the problem of poor long sequence modeling ability of RNN, LSTM can only capture the positive semantic information of the text. Therefore, to capture the reverse semantic information and realize the comprehensive modeling of the context, this paper uses the BiLSTM model. The design concept is to capture the information between the past and the future simultaneously. BiLSTM can be considered as a superposition of two layers of LSTM, its output state h_t^c at a certain time t is determined by the forward LSTM output $\overrightarrow{h_t^c}$ and the reverse LSTM output $\overleftarrow{h_t^c}$. In this model, we connect $\overrightarrow{h_t^c}$ and $\overleftarrow{h_t^c}$ to form the output of the BiLSTM layer:

$$h_t^c = \left[\overrightarrow{h_t^c}, \overleftarrow{h_t^c}\right] \tag{14}$$

The statement text representation $h_t^c(1, 2, \ldots, n)$ obtained after BiLSTM is sent to the attention mechanism layer to obtain the final representation h of the statement. The calculation process is as follows:

$$u_t = \tanh\left(W_u h_t^c + b_u\right) \tag{15}$$

$$\alpha_t = \frac{\exp(u_t)}{\sum_{t=1}^{l} \exp(u_t)} \tag{16}$$

$$v = \sum_{i=1}^{l} \alpha_t h_t \tag{17}$$

$$h = \tanh\left(W_v v + W_h h_t^c\right) \tag{18}$$

Here, h_t^c is the text representation of the input to the attention mechanism. Firstly, the implicit representation u_t of h_t^c is obtained, α_t represents the weight corresponding to u_t, W_u and b_u are the learnable weights and biases, respectively.

The feature vector h obtained through the attention mechanism layer is input into the output layer to reduce the dimension. The vector with the output dimension as the number of categories of emotional labels is obtained through the fully connected layer. Finally, the output results are normalized by Soft max to get the desired emotional polarity results.

4 Experiments

Datasets: To thoroughly test the effect and generalization ability of the model, this paper selects three data sets for experiments. The public data sets are selected from weibo_senti_100k and NLPCC14, among which the weibo_senti_100k data set is selected from Weibo. Because Weibo limits comments to 140 words, the data characteristics of the data set are short text and strong colloquialisms, which can test the effect of the model's short text sentiment classification. It contains 119988 data with emotional tags, including 59994 positive and negative comment texts. The NLPCC14 dataset contains a total of 10,000 data with emotional tendencies, of which 5000 are positive and 5000 are negative. The dataset is characterized by moderate text length and mostly reviews for goods, which can test the modeling effect of the model on multi-sentence text. In addition, we also grabbed some review data from the reviews of Meituan, Dianping and Ctrip.com to form a business review data set MDC22 to test the generalization ability of the model further. After data cleaning and deduplication, we removed invalid data such as short comments and spam comments to obtain a total of 26435 review data and then obtained 13195 positive and 13240 negative texts by manually labeling emotional tendencies. Before training, we disrupted the data order of the dataset, deleted some symbols such as '#,' '\' and other contents unrelated to emotional expression, and performed data preprocessing operations such as stop word removal. The processed data examples are shown in Table 1, where the label value 1 indicates a positive tendency. Data samples for the dataset are shown in Table 2.

Table 2: Preprocessed data samples for the dataset

Label	Comments
1	Buy three, just buy told him, he said, I know his heart too much.[applaud] [titter]
0	[Surprised] Husband said I'd degenerate into this if I were lazy anymore.[disappointed] [disappointed]
1	Explain very well but do have a particular C++ foundation and experience. Otherwise, it is difficult to understand and needs to be read several times.
0	Not as good as before, the environment is not as good as before, the sofa is not clean, but the yogurt is not bad, milk tea is not as good as before.
0	I don't know why a western restaurant puts on such loud music. Does the leader think it's better?
0	Yesterday's booked tickets to Urumqi. Today I found that the price of more than 160, you become very fast!
1	Stay at this hotel on every business trip. The environment can also be mainly cheap and easy to reimburse. I can recommend [praise].

Experimental Parameter Settings: The model mentioned in the text is completed on the basis of Pytorch1.11, Transformers 4.16.2 framework and python3.8 environment. The pre-training model uses the PERT model. The dimension of the word vector is set to 768. The parameter of Multi-Head-Attention is 12, and the learning rate is 2e-5. Epochs are the training ground for the training set, which is set to 20 here. Batch size is the number of training samples sent to the neural network each time. The above three data sets are set to 16, 32 and 32, respectively. The maximum length of the text Max length is set to 150. The number of hidden units of the BiLSTM network is 128, and the number of layers is 2. The model training uses the Adam optimizer [33] and uses the 'Early Stopping' method

in the training phase. The dropout is set to 0.2. The parameter settings of the pre-training model are shown in Table 3.

Table 3: Experimental parameter settings

Parameter	Value
Embedding size	768
Batch size	16、32、32
Epoch	20
Max length	150
Learning rate	2e-5
Optimizer	Adam
Dropout	0.2

Evaluation Metrics: The evaluation indexes involved in this paper are precision rate P, recall rate R, F1 value and accuracy rate A. The above evaluation indicators are defined as follows

$$P = \frac{T_P}{T_P + F_P} \tag{19}$$

$$R = \frac{T_P}{T_P + F_N} \tag{20}$$

$$F1 = \frac{2PR}{P + R} \tag{21}$$

$$A = \frac{T_P + T_N}{T_P + F_N + T_N + F_P} \tag{22}$$

Among them, T_P represents the number of samples predicted to be positive in the positive emotion samples, F_P represents the number of samples predicted to be positive in the negative emotion samples, F_N represents the number of samples predicted to be negative in the positive emotion samples, and T_N represents the number of samples predicted to be negative in the negative emotion samples. The confusion matrix is shown in Table 4.

Table 4: Confusion matrix

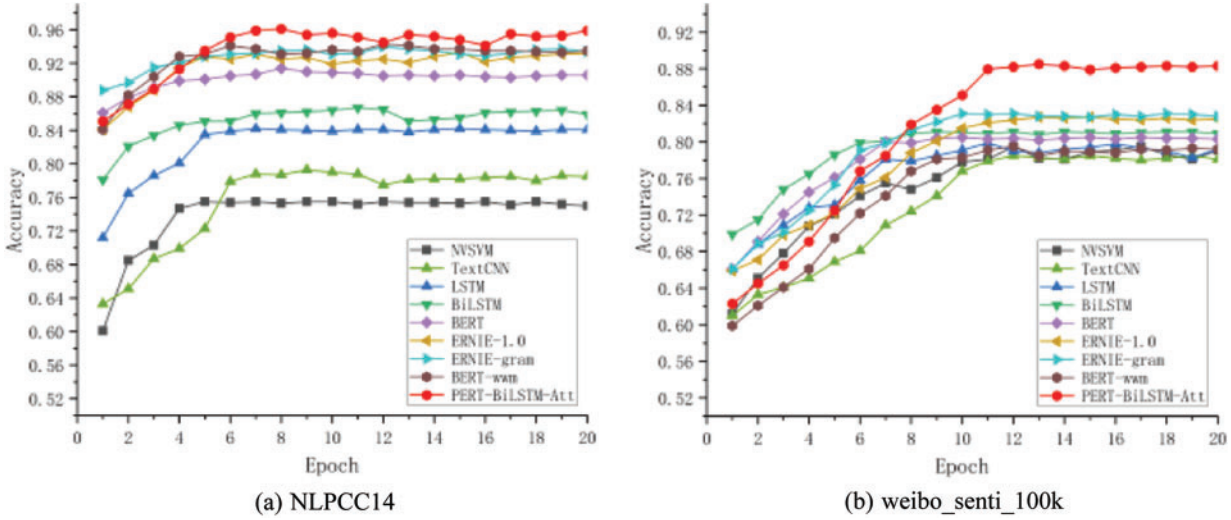
Prediction class	Actual class	
	Positive direction	Negative direction
Positive direction	Actual positive and predicted positive(T_P)	Actual negative but predicted positive(F_P)
Negative direction	Actual positive but predicted negative(F_N)	Actual negative and predicted negative(T_N)

4.1 Performance Comparison

To verify the effectiveness of the proposed method and verify the impact of the combination of each module of the model on the experimental results, this paper sets up three sets of comparative experiments. The first group of experiments directly uses the commonly used word vector model and the pre-training model to classify emotions directly. In this group of comparative experiments, the text vector obtains the text feature representation through the word vector model or the pre-training model. The obtained text features are directly input into the Softmax classifier for classification after dimension reduction through the fully connected layer. The models involved are as follows. The experimental results are shown in Table 5. Figs. 5a, b and c shows how the accuracy of the three data sets used in this experiment varies with the number of training rounds.

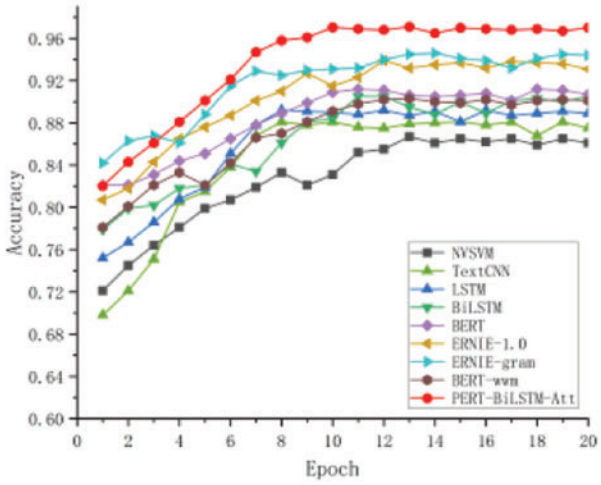
Table 5: Results of the comparative experiment (Basic model)

Model	NLPCC14		weibo_senti_100k		MDC22	
	Accuracy	F1	Accuracy	F1	Accuracy	F1
NBSVM	79.55	79.49	86.73	86.73	75.56	75.56
TextCNN	78.46	78.47	88.12	88.14	78.88	78.81
LSTM	79.89	79.91	89.28	89.21	84.23	84.23
BiLSTM	81.11	81.23	90.52	90.52	86.52	86.52
BERT	80.48	80.41	91.17	91.22	90.47	90.47
ERNIE-1.0	82.65	82.66	93.84	93.82	92.86	92.81
ERNIE-gram	83.13	82.97	94.53	94.48	93.23	93.22
BERT-wwm	79.56	79.62	90.22	90.26	93.73	93.74
PERT-BiLSTM-Att	**88.56**	**88.59**	**97.05**	**96.99**	**95.95**	**95.95**



(a) NLPCC14 (b) weibo_senti_100k

Figure 5: (Continued)



(c) MDC22

Figure 5: Results of comparative experiment (Basic model)

NBSVM [34]: SVM with Naive Bayes (NBSVM) is a classification model of machine learning. The word vector is initialized by Global Vectors for Word Representation (Glove) [35]. Then each word vector of the text is added, and the mean value is input into the classifier for classification.

TextCNN: It is composed of three convolution kernels of different sizes. The text is convoluted and pooled directly through the convolution neural network, and the final output is the result.

LSTM: A variant of the recurrent neural network RNN, which outputs the results by obtaining the state of the last moment of the statement.

BiLSTM: Bidirectional LSTM, the output is composed of the final state of the forward LSTM and the reverse LSTM.

BERT-base: A pre-training model proposed by DevlinJ et al. that uses a self-attention mechanism for training.

ERNIE [36]: Enhanced representation through knowledge integration (ERNIE1.0) proposed by Baidu. The model is trained from the social media corpus and improves the MASK method during training so that the model can learn more information.

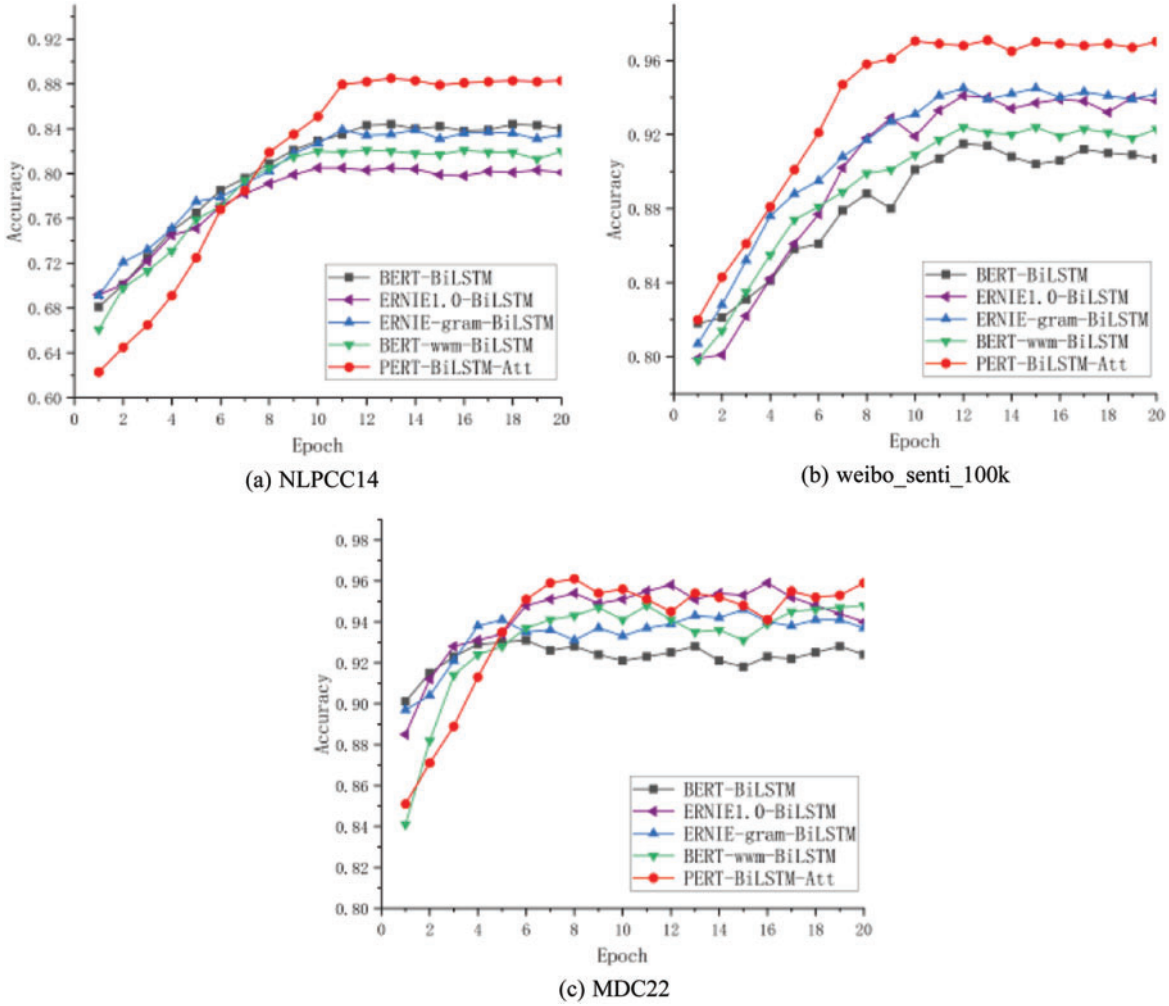
ERNIE-gram [37]: Based on ERNIE1.0, a two-stream structure is adopted to realize single-position multi-semantic granularity prediction in the pre-training process.

BERT-wwm: BERT-Whole Word Masking model based on full-word masking released by IFLYTEK Joint Laboratory.

The second group of comparative experiments uses several commonly used pre-training models to process text to obtain text feature vectors and then inputs the obtained text features into BiLSTM for sentiment classification. The experimental results are shown in Table 6. Figs. 6a, b and c shows how the accuracy of the three data sets used in this experiment varies with the number of training rounds.

Table 6: Results of the comparative experiment (Basic model+BiLSTM)

Model	NLPCC14		weibo_senti_100k		MDC22	
	Accuracy	F1	Accuracy	F1	Accuracy	F1
BERT-BiLSTM	84.28	84.31	91.47	91.45	92.40	92.41
ERNIE-1.0-BiLSTM	80.54	80.57	93.99	93.97	95.20	95.22
ERNIE-gram-BiLSTM	83.85	82.88	94.49	94.45	93.03	92.99
BERT-wwm-BiLSTM	82.19	82.17	92.44	92.43	94.21	94.22
PERT-BiLSTM-Att	**88.56**	**88.59**	**97.05**	**96.99**	**95.95**	**95.95**



(a) NLPCC14 (b) weibo_senti_100k

(c) MDC22

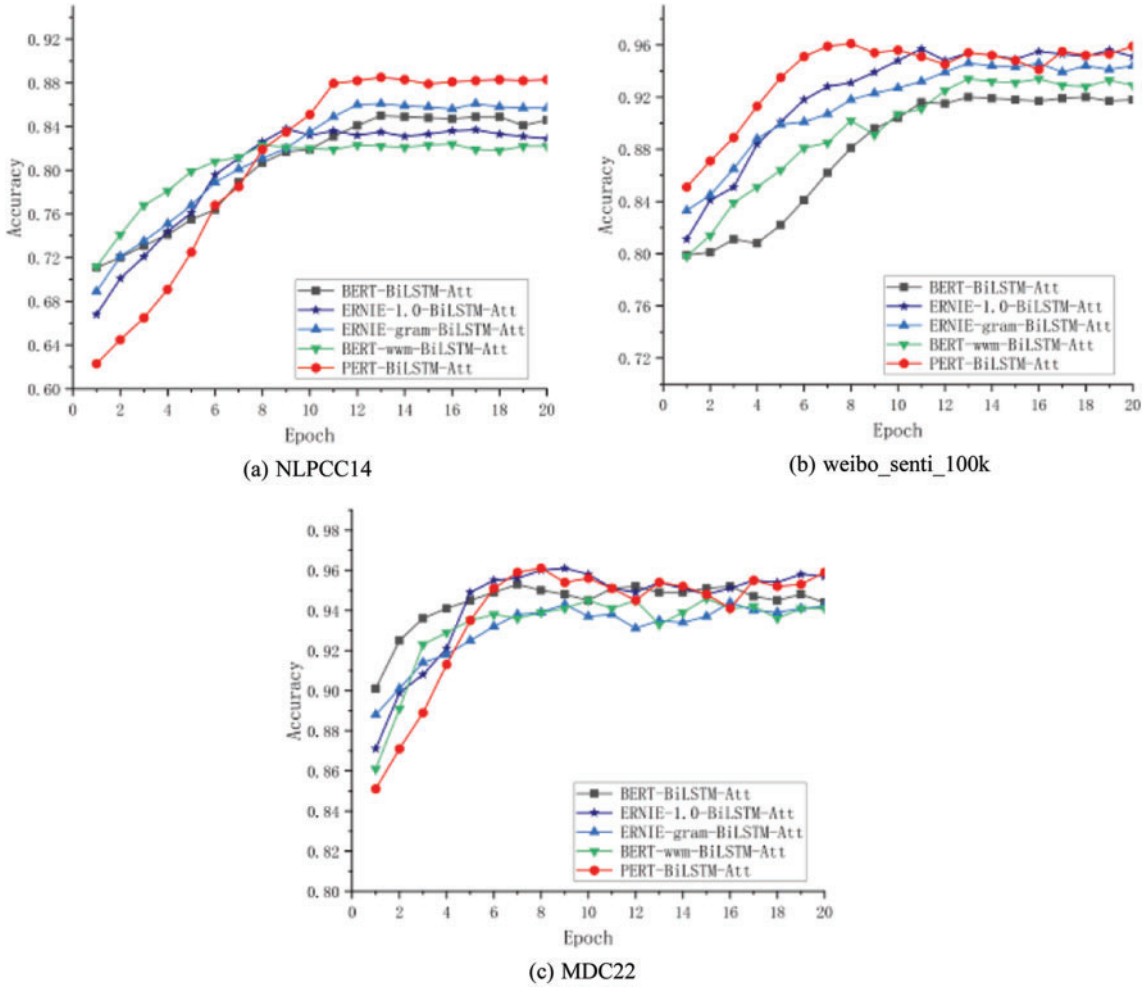
Figure 6: Results of the comparative experiment (Basic model+BiLSTM)

On the basis of the second group of experiments, the third group of comparative experiments inputs the word order information output from the BiLSTM layer into the attention mechanism layer to realize the attention to the words with a greater contribution. Finally, the output probability is

normalized by dimensionality reduction and Softmax. The experimental results are shown in Table 7. Figs. 7a, b and c shows how the accuracy of the three data sets used in this experiment varies with the number of training rounds.

Table 7: Results of the comparative experiment (Basic model+BiLSTM+Att)

Model	NLPCC14		weibo_senti_100k		MDC22	
	Accuracy	F1	Accuracy	F1	Accuracy	F1
BERT-BiLSTM-Att	84.99	85.01	92.00	92.03	94.70	94.71
ERNIE-1.0-BiLSTM-Att	83.77	83.77	95.72	95.72	95.33	95.34
ERNIE-gram-BiLSTM-Att	85.98	85.97	94.69	94.71	93.07	93.05
BERT-wwm-BiLSTM-Att	82.25	82.24	93.38	93.38	94.00	93.98
PERT-BiLSTM-Att	**88.56**	**88.59**	**97.05**	**96.99**	**95.95**	**95.95**



(a) NLPCC14 (b) weibo_senti_100k

(c) MDC22

Figure 7: Results of the comparative experiment (Basic model+BiLSTM+Att)

4.2 Experimental Result Analysis

It can be seen from Table 5 that compared with the commonly used word vector model and pre-training model, the PERT-BiLSTM-Att model proposed in this paper has a better sentiment classification effect on the three data sets than other models. Especially for the classification of the NLPCC14 data set, the accuracy and F1 value of the model proposed in this paper are more than 2% higher than other models, which fully demonstrates the learning modeling ability of the model for long text data with the multi-sentence combination. From Tables 6 and 7, it can be seen that the classification effect of the model is improved to varying degrees by adding the BiLSTM layer and attention layer on the basis of the existing commonly used pre-training model, indicating that BiLSTM and attention layer strengthens the model's attention to the overall structure of the text, capture the keywords in the text through the attention mechanism, and comprehensively judge the emotional tendency of the text from the two levels of words and sentences.

Through the above graph of accuracy with the change of training rounds, we can find that the effect of our proposed model on the three data sets is better than that of the comparison model. We can also find that for data sets such as NLPCC14, which have a large number of short text data and a small amount of data, our proposed model and other models need multiple rounds of training models to converge and achieve better accuracy. This also shows that the current commonly used models have insufficient modeling ability for short text and small data sets, which also verifies that we try to make up for this defect by using the attention mechanism.

From the overall experimental results, the model proposed in this paper combines the PERT pre-training model's learning of the word's position information to avoid the noise problem caused by the traditional pre-training model through word masking training. Through the BiLSTM modeling of the front and back time series and the introduction of the attention mechanism for the semantic learning of keywords, the model has a good classification effect for all types of data sets, showing the good generalization performance of the model and fully demonstrating the effectiveness of the model.

4.3 Ablation Experiment

To thoroughly verify the role of each module of our proposed model, this ablation experiment was designed.

From the results of the ablation experiment Table 8, it can be seen that if the pre-training layer is removed, only BiLSTM is used to extract context information. Then the attention mechanism is used to obtain keywords, the model has a certain improvement compared with the classification effect of only BiLSTM in Table 5, indicating that the attention mechanism has a positive effect on extracting keywords to assist sentence polarity prediction. At the same time, it also shows that if the pre-training model is missing for the dynamic expression of word vectors, the model effect will decrease. If the attention layer is removed, the model degenerates into PERT-BiLSTM. It can be seen from the results that if the attention layer is removed, the classification effect for the three data sets is reduced to varying degrees, indicating that adding the attention mechanism to distinguish the importance of each word can effectively improve the model effect. Furthermore, if the attention layer and the BiLSTM layer are removed, only the pre-training model is used to connect the fully connected layer for sentiment classification. The effect will be further reduced, which indicates that it is necessary to add the BiLSTM layer to strengthen the learning ability for long sequences.

In this paper, the BiLSTM layer is used to enhance the model's ability to model long sequences. To verify the effect of BiLSTM, we try to use other models to replace BiLSTM. Our experiment uses the gated recurrent unit (GRU), the bidirectional gate recurrent unit (BiGRU), and a unidirectional

LSTM to replace the BiLSTM layer of the model. The experimental results are shown in Table 9. The performance of the BiLSTM model is better than other models. We also try to remove the BiLSTM layer to observe the experimental results. Compared with the data in Table 8, the accuracy of the model on the weibo_senti_100k and MDC22 datasets has been greatly reduced after removing the BiLSTM layer, which further illustrates the importance of the BiLSTM layer.

Table 8: Ablation experiments to verify the role of each module

Model	NLPCC14		weibo_senti_100k		MDC22	
	Accuracy	F1	Accuracy	F1	Accuracy	F1
PERT	83.71	83.78	94.50	94.50	94.81	94.85
BiLSTM-Att	84.56	84.56	92.52	92.52	84.21	84.23
PERT-BiLSTM	87.79	87.78	94.58	94.56	94.88	95.01
PERT-BiLSTM-Att	**88.56**	**88.59**	**97.05**	**96.99**	**95.95**	**95.95**

Table 9: Ablation experiment to verify the effect of BiLSTM layer

Model	NLPCC14		weibo_senti_100k		MDC22	
	Accuracy	F1	Accuracy	F1	Accuracy	F1
PERT-Att	85.23	85.45	92.88	92.95	93.59	93.77
PERT-GRU-Att	85.66	85.98	92.22	92.24	93.33	93.14
PERT-BiGRU-Att	86.93	86.22	94.81	94.79	93.97	93.91
BERT-LSTM-Att	87.25	86.96	94.78	94.32	94.77	94.21
PERT-BiLSTM-Att	**88.56**	**88.59**	**97.05**	**96.99**	**95.95**	**95.95**

5 Conclusion

Based on the pre-training model PERT of the out-of-order language model, the bidirectional long-term and short-term memory network BiLSTM and the attention mechanism Att, this paper proposes the PERT-BiLSTM-Att model to analyze the sentiment of online public opinion texts. The text's vector representation is obtained using the pre-training model PERT obtained by training the location information based on a large amount of corpus, which strengthens the expression of the location information of the word. The BiLSTM enhances the modeling ability for long sequences and then strengthens the proportion of keywords through the attention mechanism to further enhance the accuracy of classification. Experimental results show that compared with other commonly used sentiment analysis models, our proposed model can achieve better classification results on three data sets, which proves the generalization effect of the model.

In the next step, we will further improve the model, such as introducing more fine-grained sentiment analysis methods to classify texts more effectively. In addition, due to the short and colloquial characteristics of the network public opinion text, other semantic features contained in it will be further explored, such as the included emoticons, topics and semantic subjects to further improve the accuracy of classification.

Funding Statement: This work was partially supported by the Chongqing Natural Science Foundation of China(Grant No. CSTB2022NSCQ-MSX1417), the Science and Technology Research Program of Chongqing Municipal Education Commission(Grant No. KJZD-K202200513) and Chongqing Normal University Fund (Grant No. 22XLB003).

Conflicts of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

[1] J. Wang, Y. Peng, Y. Zhao and J. Yang, "Survey of social network public opinion information extraction based on deep learning," *Computer Science*, vol. 49, no. 8, pp. 279–293, 2022.
[2] S. Peng, A. Zhou, S. Liao, Y. Zhou, D. Liu *et al.,* "Forecast method of public opinion evolution based on graph attention network," *Journal of Sichuan University (Natural Science Edition)*, vol. 59, no. 1, pp. 13004, 2022.
[3] J. Devlin, M. W. Chang, K. Lee and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint, arXiv: 181004805, 2018.
[4] Y. Cui, Z. Yang and T. Liu, "Pert: Pre-training bert with permuted language model," arXiv preprint arXiv: 2203.06906, 2022.
[5] S. Liu, J. Zhao, H. Yang and G. Xu, "A survey of sentiment analysis," *Software Guide*, vol. 17, no. 6, pp. 4–21, 2022.
[6] Z. Yanyan, Q. Bing, S. Qiuhui and L. Ting, "Large-scale sentiment lexicon collection and its application in sentiment classification," *Journal of Chinese Information Processing*, vol. 31, no. 2, pp. 187–193, 2017.
[7] G. Song, D. Cheng, S. Zhang, W. Liu and X. Ding, "A model of textual emotion score calculation based on the emotion dictionary," *China Computer & Communication*, vol. 33, no. 22, pp. 56–62, 2021.
[8] V. N. Vapnik, "A note on one class of perceptrons," *Automat. Rem. Control*, vol. 25, pp. 821–837, 1964.
[9] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.
[10] B. Pang, L. Lee and S. Vaithyanathan, "Thumbs up? sentiment classification using machine learning techniques," arXiv preprint cs/0205070, 2002.
[11] J. Zhu, J. Liu, T. Zhang and L. Qiu, "Sentiment polarity classification method based on sentiment dictionary an ensemble learning," *J. Comput. Appl*, vol. 6, no. 15, pp. 95–98, 2018.
[12] A. Graves, "Generating sequences with recurrent neural networks," arXiv preprint arXiv: 1308.0850, 2013.
[13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[14] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares *et al.,* "Learning phrase representations using rnn encoder-decoder for statistical machine translation," arXiv preprint arXiv: 1406.1078, 2014.
[15] Y. Kim, "Convolutional neural networks for sentence classification," Arxiv preprint arxiv: 14085882, pp. D14–1181, 2014. [Online]. Available: https://doi.org/10.3115/v1
[16] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in Neural Information Processing Systems*, vol. 26, pp. 3111–3119, 2013.
[17] D. Yongping, Z. Xiaozheng and P. Bingbing, "Short text sentiment classification based on cnn-lstm model," *Journal of Beijing University of Technology*, vol. 45, no. 7, pp. 662–670, 2019.
[18] H. Yin, S. Li, G. Zhengxian and G. Zhou, "Imbalanced emotion classification based on multi-channel lstm," *Journal of Chinese Information Processing*, vol. 32, no. 1, pp. 139–145, 2018.
[19] V. Mnih, N. Heess and A. Graves, "Recurrent models of visual attention," *Advances in Neural Information Processing Systems*, vol. 27, pp. 2204–2212, 2014.

[20] D. Bahdanau, K. Cho and Y. Bengio, "Neural machine translation by jointly learning to align and translate," arXiv preprint arXiv: 1409.0473, 2014.
[21] Z. Lin, M. Feng, C. N. d. Santos, M. Yu, B. Xiang *et al.,* "A structured self-attentive sentence embedding," arXiv preprint arXiv: 1703.03130, 2017.
[22] G. Xu, J. Zhao and H. Yang, "A review of text feature extraction methods," *Software Guide*, vol. 17, no. 5, pp. 13–18, 2018.
[23] A. Martins and R. Astudillo, "From softmax to sparsemax: A sparse model of attention and multi-label classification," in *Int. Conf. on Machine Learning*, New York City, NY, USA, PMLR, pp. 1614–1623, 2016.
[24] Y. Kim, C. Denton, L. Hoang and A. M. Rush, "Structured attention networks," arXiv preprint arXiv: 1702.00887, 2017.
[25] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola *et al.,* "Hierarchical attention networks for document classification," in *Proc. of the 2016 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego City CA, USA, pp. 1480–1489, 2016.
[26] J. Cheng, L. Dong and M. Lapata, "Long short-term memory-networks for machine reading," arXiv preprint arXiv: 1601.06733, 2016.
[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones *et al.,* "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, pp. 6000–6010, 2017.
[28] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi *et al.,* "Google's neural machine translation system: Bridging the gap between human and machine translation," arXiv preprint arXiv:1609.08144, 2016.
[29] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li *et al.,* "Residual attention network for image classification," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Honolulu City, HI, USA, pp. 3156–3164, 2017.
[30] J. L. Ba, J. R. Kiros and G. E. Hinton, "Layer normalization," arXiv preprint arXiv: 1607.06450, 2016.
[31] Y. Cui, W. Che, T. Liu, B. Qin and Z. Yang, "Pre-training with whole word masking for Chinese bert," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3504–3514, 2021.
[32] X. Shi, Z. Chen, H. Wang, D. -Y. Yeung, W. -K. Wong *et al.,* "Convolutional lstm network: A machine learning approach for precipitation nowcasting," *Advances in Neural Information Processing Systems*, vol. 28, pp. 802–810, 2015.
[33] A. KingaD, "A method for stochastic optimization," in *Anon. Int. Conf. on Learning Representations*, San Diego City, CA, USA, SanDego: ICLR, 2015.
[34] S. I. Wang and C. D. Manning, "Baselines and bigrams: Simple, good sentiment and topic classification," in *Proc. of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Jeju Island, Kr, pp. 90–94, 2012.
[35] J. Pennington, R. Socher and C. D. Manning, "Glove: Global vectors for word representation," in *Proc. of the 2014 Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, Doha City, Qatar, pp. 1532–1543, 2014.
[36] Y. Sun, S. Wang, Y. Li, S. Feng, X. Chen *et al.,* "Ernie: Enhanced representation through knowledge integration," arXiv preprint arXiv: 1904.09223, 2019.
[37] D. Xiao, Y. -K. Li, H. Zhang, Y. Sun, H. Tian *et al.,* "Ernie-gram: Pre-training with explicitly n-gram masked language modeling for natural language understanding," arXiv preprint arXiv: 2010.12148, 2020.