electronics

MDPI

Article

Sentiment Analysis of Comment Data Based on BERT-ETextCNN-ELSTM

Lujuan Deng, Tiantian Yin *, Zuhe Li and Qingxia Ge

School of Computer and Communication Engineering, Zhengzhou University of Light Industry,
Zhengzhou 450002, China; lujuandeng@163.com (L.D.); zuheli@zzuli.edu.cn (Z.L.);
332007040514@email.zzuli.edu.cn (Q.G.)
* Correspondence: 332007040532@email.zzuli.edu.cn

Abstract: With the rapid popularity and continuous development of social networks, users' communication and interaction through platforms such as microblogs and forums have become more and more frequent. The comment data on these platforms reflect users' opinions and sentiment tendencies, and sentiment analysis of comment data has become one of the hot spots and difficulties in current research. In this paper, we propose a BERT-ETextCNN-ELSTM (Bidirectional Encoder Representations from Transformers–Enhanced Convolution Neural Networks–Enhanced Long Short-Term Memory) model for sentiment analysis. The model takes text after word embedding and BERT encoder processing and feeds it to an optimized CNN layer for convolutional operations in order to extract local features of the text. The features from the CNN layer are then fed into the LSTM layer for time-series modeling to capture long-term dependencies in the text. The experimental results proved that compared with TextCNN (Convolution Neural Networks), LSTM (Long Short-Term Memory), TextCNN-LSTM (Convolution Neural Networks–Long Short-Term Memory), and BiLSTM-ATT (Bidirectional Long Short-Term Memory Network–Attention), the model proposed in this paper was more effective in sentiment analysis. In the experimental data, the model reached a maximum of 0.89, 0.88, and 0.86 in terms of accuracy, F1 value, and macro-average F1 value, respectively, on both datasets, proving that the model proposed in this paper was more effective in sentiment analysis of comment data. The proposed model achieved better performance in the review sentiment analysis task and significantly outperformed the other comparable models.

Keywords: sentiment analysis; BERT; long short-term memory; convolutional neural network

check for updates

Citation: Deng, L.; Yin, T.; Li, Z.; Ge, Q. Sentiment Analysis of Comment Data Based on BERT-ETextCNN-ELSTM. *Electronics* **2023**, *12*, 2910. https://doi.org/10.3390/electronics12132910

Academic Editor: Yu-an Tan

Received: 22 March 2023
Revised: 16 June 2023
Accepted: 28 June 2023
Published: 3 July 2023

1. Introduction

With the rapid development and popularity of social media platforms such as Weibo, Zhihu, and Twitter [1–3], more and more users can post their views, attitudes, and emotions on certain topics on these social media platforms, resulting in a large amount of textual data consisting of comments with emotional overtones. Analyzing textual data with emotional overtones not only makes it possible to obtain information about the user's psychological state at the moment, his or her inclination to voice an opinion on various matters, and to understand the general views and attitudes of users, but the data also have potential economic value [4]. The analysis can even be used to monitor undesirable comments and thus ensure online safety. Therefore, sentiment analysis of text comment data has important research implications.

The three main methods for text sentiment analysis are based on sentiment dictionaries, machine learning, and deep learning [5]. The sentiment dictionary approach matches a dataset with words in a sentiment dictionary. It calculates the sentiment polarity of the text through weighting, but a complete dictionary is challenging to construct [6]. Machine learning [7] methods use algorithms such as Naive Bayes (NB) and Support Vector Machines (SVM) to achieve sentiment analysis. Still, traditional machine learning methods often fail to integrate contextual information, thoroughly affecting the accuracy of classification, so they are not well suited to a variety of scenarios. Both methods have apparent drawbacks, based on which deep learning-based approaches have been proposed [8]. Compared with traditional machine learning models, deep learning methods can actively extract text features [9–15], reduce the complexity of text construction features, and perform better on sentiment analysis tasks. This paper focuses on sentiment analysis using deep learning methods.

Typical neural network learning methods include Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) networks, etc. Sentiment analysis methods based on deep learning can be subdivided into single neural network sentiment analysis methods, hybrid (combined, fused) neural network sentiment analysis methods, sentiment analysis with the introduction of attention mechanisms, and sentiment analysis using pre-trained models. This paper uses pre-trained models and optimized hybrid (combinatorial, fusion) neural networks for sentiment analysis to effectively address the problem of ignoring contextual semantics in traditional sentiment analysis methods and to better extract the semantic information of the corresponding words to achieve effective sentiment classification of text.

2. Related Studies

Sentiment analysis is an important research hotspot in the field of natural language processing and has a wide range of research areas in data mining, web mining, text mining, and opinion analysis. In recent years, sentiment analysis methods based on deep learning have been widely used, the most common of which are convolutional neural network [16] models and recurrent neural network models.

With the continuous development of deep learning technology, more and more researchers have started to apply deep learning research methods to sentiment analysis of text classes. For example, convolutional neural networks and recurrent neural networks have been widely used by Sun [17] and others have used recurrent neural networks to process text features in order to address the problem of sparse text features, achieving good results on Chinese datasets. However, because of the special structure of RNNs, gradient explosion and gradient dispersion problems are prone to occur. Therefore, variants of RNNs are generally used to deal with sentiment analysis problems at present.

Alhagr et al. [18] argued that sentiment analysis is essentially a sequence problem and so they used a Long Short-Term Memory network (LSTM) to deal with sequences and proposed six LSTM models with different parameters. These models have shown excellent performance on multiple datasets. However, it is difficult to accurately capture the local information of a sentence using only LSTM models, so some researchers have also explored combining deep learning methods such as CNN and LSTM to improve the accuracy of sentiment analysis.

The convolutional neural network model proposed by Kim [19] is one of the classic approaches in the field of sentiment analysis. The model used convolutional and max-pooling operations to extract features from the input text and fed the extracted features into a fully connected layer for classification. Kim applied the model to an IMDB movie review dataset and achieved the best performance at the time.

The recurrent neural network-based sentiment analysis method proposed by Zhuge et al. [20] in 2015 used a Long Short-Term Memory network model (LSTM) to encode text and then used a word vector and sentiment dictionary approach for text feature extraction. The method was applied to several datasets and achieved good performance.

Zhou et al. [21] proposed a deep learning-based sentiment analysis method, the Bidirectional Long Short-Term Memory Network (BiLSTM) model, for text encoding and an attention mechanism to adaptively select important text features. In sentiment analysis tasks, the model could accurately identify sentiment tendencies in text. In addition, the model had good generalization capabilities and could be applied to different datasets and tasks.

Later, Cheng et al. [22] proposed a method for simultaneous text reading comprehension and aspect-level-based sentiment analysis. The method used a Gated Recurrent Unit (GRU) to encode the text and a Multi-Head Attention mechanism to adaptively select the important features in the text. In addition, the method could simultaneously identify different aspects of the text and perform sentiment analysis separately, thus improving the accuracy and efficiency of sentiment analysis.

Munikar et al. [23] used a deep bidirectional language model based on the Transformer architecture, a pre-trained BERT model, and fine-tuned it. Their experiments showed that their model outperformed other popular models without the complex architecture.

Based on the above summary comparison, in the field of sentiment analysis [24], deep learning methods that have been developed in recent years [25–27] can automatically and quickly extract relevant features from large-scale text data and capture deep semantic information more easily, with better classification results. However, there are still limitations in word vector representation and the neural network feature extraction processes in deep learning methods [28–30], which may lead to incomplete feature extraction or failure to adequately capture semantic information, thus affecting the classification results. To address this problem, this paper constructed BERT and optimized an improved CNN-LSTM model as BERT-ETextCNN-ELSTM (BERT–Enhanced Convolution Neural Networks–Enhanced Long Short-Term Memory) to improve comment sentiment analysis with improved accuracy and efficiency. While retaining the advantages of CNN and LSTM models, the model was enhanced with the introduction of BERT and optimized CNN-LSTM for representation learning and generalization, aiming to further improve the accuracy and efficiency of sentiment analysis.

3. Model Construction

The flow of the model is shown in Figure 1. In this paper, a fused BERT and optimally improved TextCNN-LSTM model were constructed as BERT-ETextCNN-ELSTM. In the model architecture, a fusion mechanism was introduced to fuse BERT, text embedding, and CNN layer representations. This fusion allowed the model to take full advantage of the deep contextual understanding of BERT and the local feature extraction capabilities of CNN. The outputs of these different layers were integrated to capture a more comprehensive representation of the input text, effectively capturing both global and local semantic information. Exploiting the synergy of the strengths of the two approaches, BERT excelled in capturing long-term dependencies and global semantic information, while CNN enhanced the model's ability to capture local nuances and fine-grained features. This fusion enabled our model to effectively capture both macro and micro levels in sentiment analysis, resulting in better performance in sentiment analysis tasks.

3.1. Input Layer

(1) Data pre-processing: the original text data are cleaned, divided into words, and deactivated to obtain a data format that can be processed by the model.

(2) Text embedding layer: The text sequence after word separation is mapped into a high-dimensional vector representation, where each word corresponds to a vector $\{W_1, W_2,..., W_{n-1}, W_n\}$, which is used to capture the semantic information of each word. In the model of this paper, a BERT [31] pre-training model was used for text embedding. The BERT model is shown in Figure 2.

3.2. Feature Extraction Layer

3.2.1. Enhanced Convolutional Neural Networks

A convolutional neural grid [19] contains convolutional layers, most commonly a two-dimensional convolutional layer. It has two spatial dimensions, height and width, which are often used to process image data, and it is currently widely used in sentiment analysis research [32–34], as shown in Figure 3. The processing of TextCNN in this paper used the Keras concatenate layer for the second part of the convolutional neural network to enhance processing and then put the second part of the six-layer convolutional neural network into the concatenate layer. Not only did this reduce the complexity of the model and loss of gradients due to model redundancy, but it also increased the number of output channels in the TextCNN network, allowing for better extraction of features from the data.

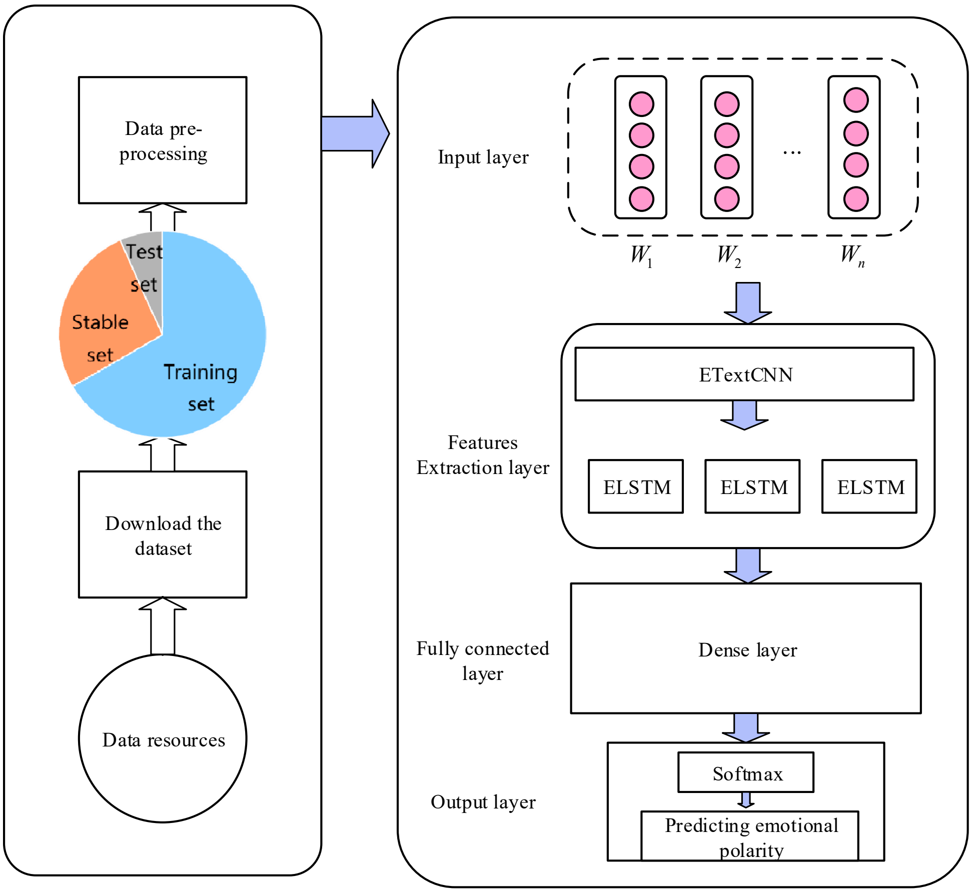

Figure 1. Flow chart of ETextCNN-ELSTM model.

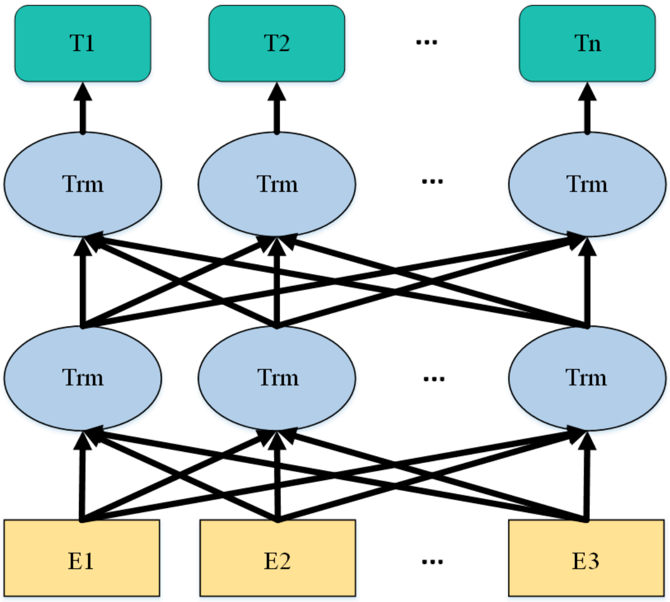

Figure 2. BERT architecture (Source: Adapted from [23]).

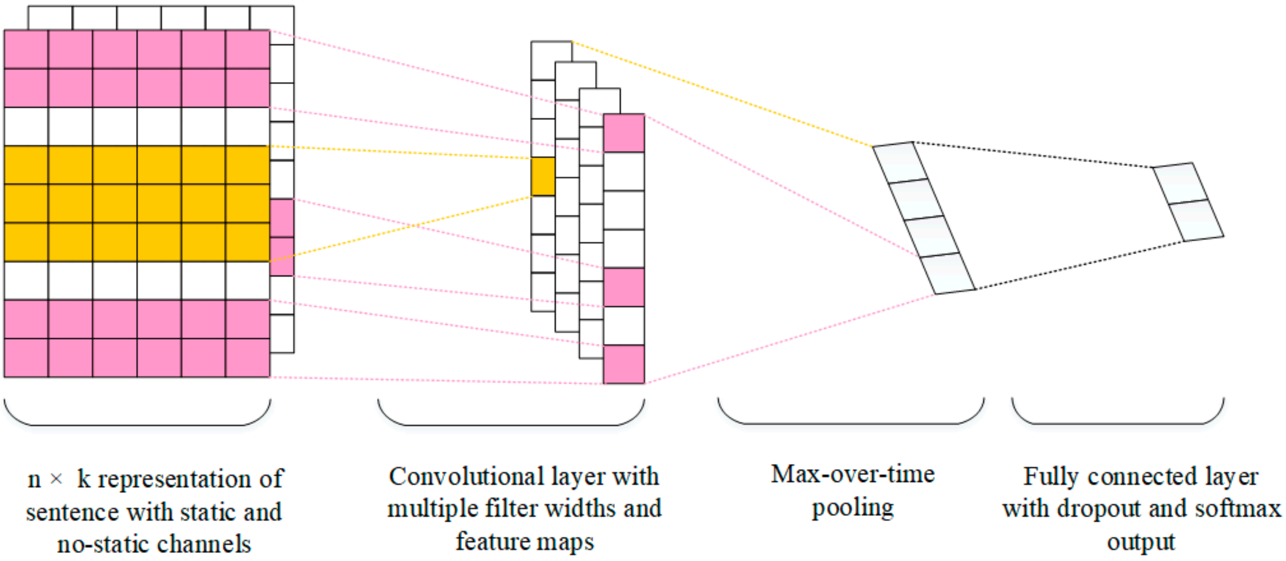
n × k representation of sentence with static and no-static channels
Convolutional layer with multiple filter widths and feature maps
Max-over-time pooling
Fully connected layer with dropout and softmax output

Figure 3. TextCNN neural network.

3.2.2. Enhanced Long and Short-Term Memory Neural Networks

In this paper, an LSTM model was considered and improved on top of the enhanced convolutional neural network. The LSTM [8] consists of oblivion, input, and output gates. The oblivion gate determines whether the information needs to be retained by the sigmoid function; the input gate filters the input information, ignores the information with the output feature dimension of 0, and updates the current cell state by combining the temporary and previous cell states; while the output gate selectively retains and ignores the information at the present moment and calculates the output result by the tanh function as the input information at the next moment. The structure of the LSTM network is shown in Figure 4, and the main calculation equations are as follows.

$$a_t = \tanh(W_a x_t + U_a h_{t-1} + b_a) \quad (1)$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (2)$$

$$f_t = e_t \circ \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (3)$$

$$o_t = \sigma(W_o x_t + U_O h_{t-1} + b_o) \quad (4)$$

$$c_t = f_t \circ c_{t-1} + i_t \circ a_t \quad (5)$$

$$h_t = o_t \circ \tanh(c_t) \quad (6)$$

where the activation function σ is a sigmoid-like function such as $\sigma(x) = (1/1 + e^{-x})$; $\circ$ is a Hadamard product operator; U and W denote the weight matrix calculated from the output h_{t-1} of the previously hidden layer and the current input x_t, respectively; and b_* is the input bias of the three S-shaped functions. In the above equations, i_t, $f_{t,}$ and o_t denote the outputs of the input, oblivion, and output gates, respectively.

In this paper, the traditional LSTM was considered to rebuild the network model as Enhanced Long Short-Term Memory (ELSTM), as shown in Figure 5. Therefore, it can be seen that this paper considered adding a fully connected layer and a dropout layer on top of the LSTM to prevent the model from overfitting in the training process. Then, the two neural networks were put into the concatenate layer to form a strengthened

LSTM neural network. Then, the three strengthened neural networks were put into the concatenate layer to enhance the LSTM neural network and achieve better extraction of data features, as shown in Figure 4. The LSTM needed to be connected to a fully connected layer to transform the output of the LSTM into the desired result. The final product of this paper was a fully connected layer of four dimensions. Based on the extracted feature vectors, the output layer used a dropout mechanism combined with softmax for sentiment classification.

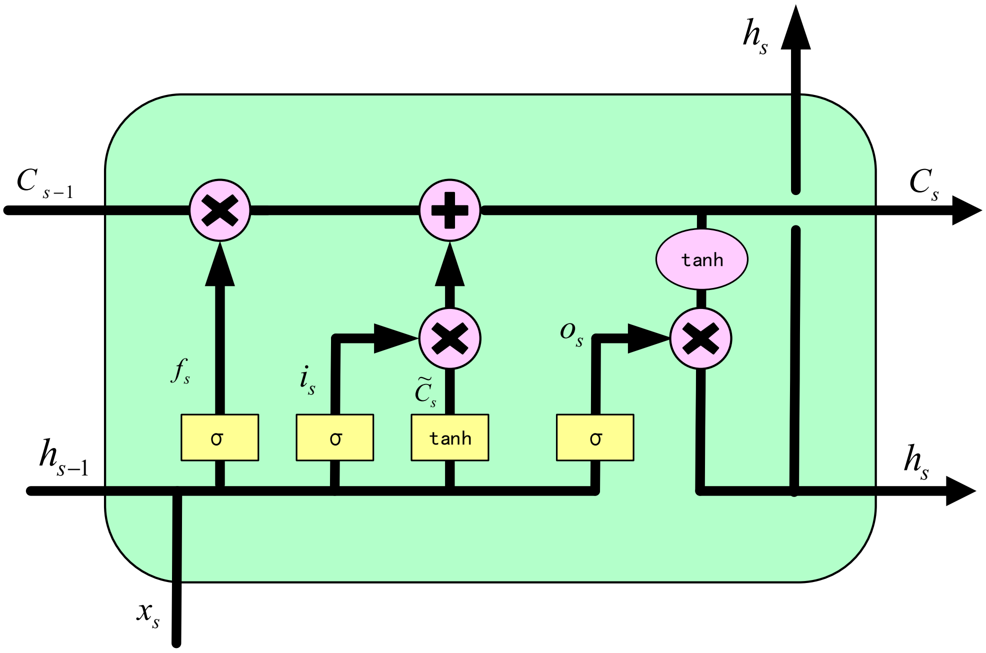

Figure 4. LSTM structure diagram (Source: Adapted from [20]).

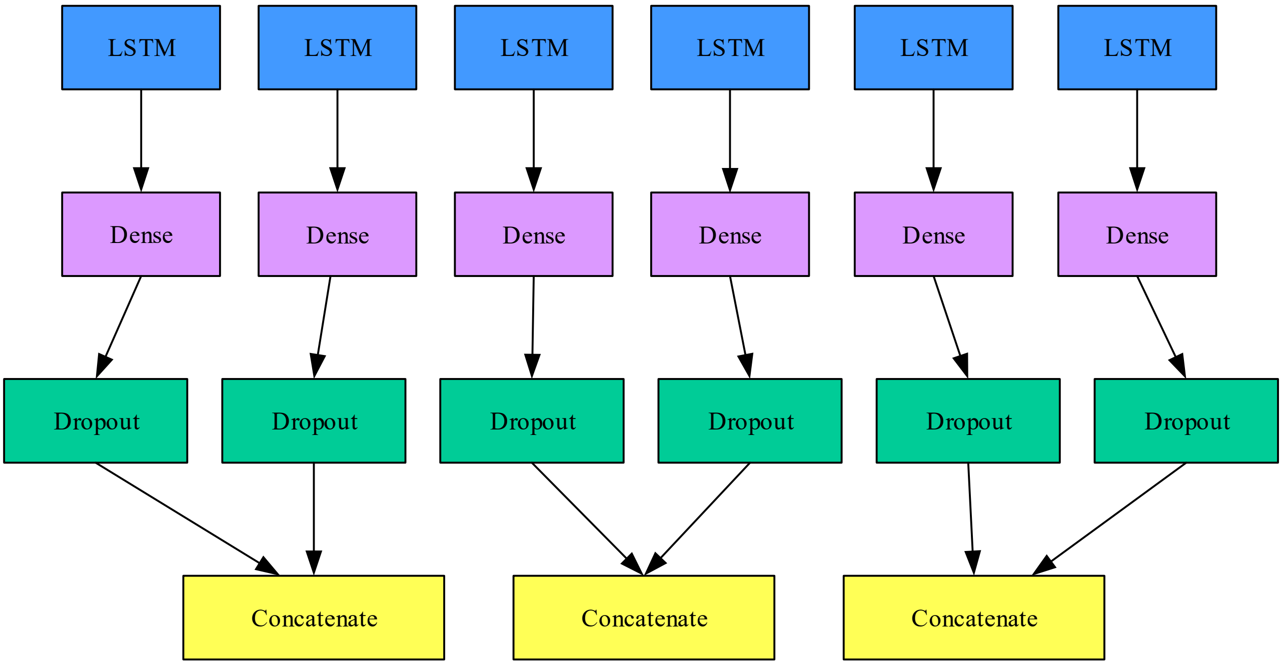

Figure 5. ELSTM structure diagram.

4. Experiment

4.1. Datasets and Pre-Processing

To more fully validate the applicability and stability of the model proposed in this paper, experiments were conducted on two Chinese datasets, namely the microblog review dataset simplifyweibo_4_moods and the hotel review dataset ChnSentiCorp_htl_all, which are described below.

The data were prepared from the official Weibo comment dataset simplifyweibo_4_moods downloaded from the web, containing four emotions: joy, anger, disgust, and depression. Each category had about 50,000 comments. The labeling methods and some of the data are shown in Tables 1 and 2. As each comment came from the web and used more symbolic language, regular expressions were applied to clean the comments. The words were split using Jieba in Python, and the length of each comment after breaking was calculated in preparation for creation of the splitter below. Figure 6 demonstrates that the number of reviews selected for each category in the chosen dataset was evenly distributed. The frequency histogram in Figure 7 shows the length of each sentence after the word splitting process, and it can be seen that the average size was 95 words and most comments were under 100 words, so the maximum number of words chosen for the next splitter was 100.

Table 1. Description of the simplifyweibo_4_moods dataset.

Field	Description
label	0 joy, 1 anger, 2 disgust, 3 depression
review	Microblog content

Table 2. Selected data from the simplifyweibo_4_moods dataset.

Serial Number	Label	Review
257031	2	It's a nasty feeling, I'm always too impulsive...
56901	0	Come and see my little pill stencil~ ~Wow, wow, wow, wow~
351395	3	The most complete one I've found. This is when you go to see your son in the north. Nostalgia. By the way, why am I wearing that torn shirt? So ugly...
249801	1	Poor, help this child to turn down, Hope will not be because of the alleged contact business what responsibility ah... is wanting fans to want crazy what situation ah? Want to...

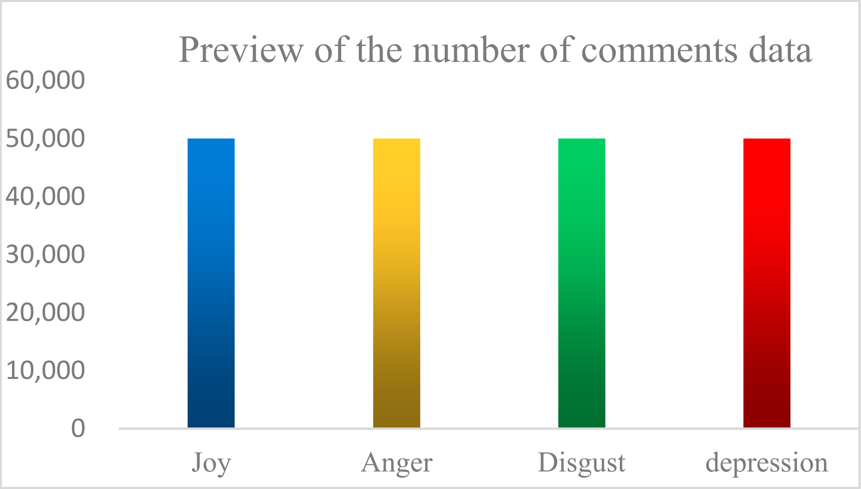

Figure 6. Preview of the number of comments in the dataset.

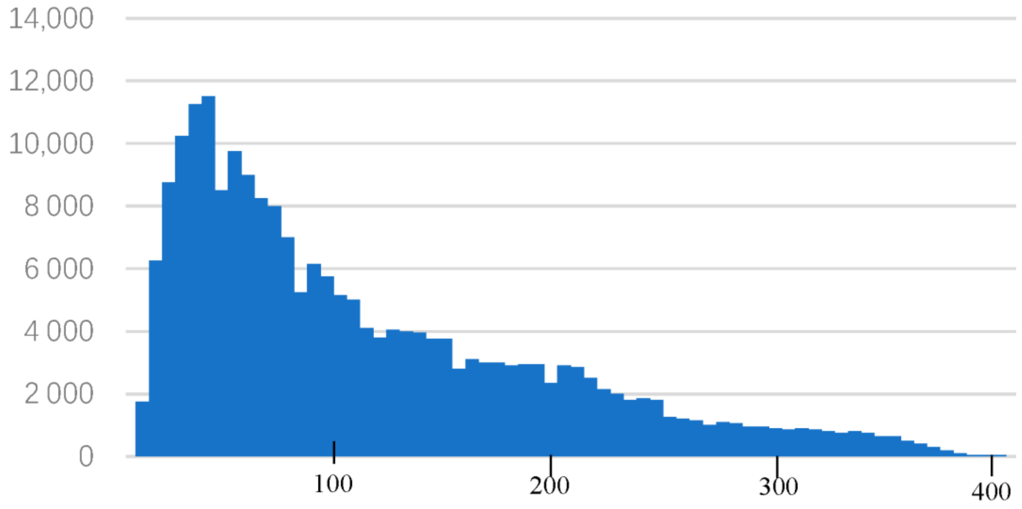

Figure 7. Preview of comment data length.

After the first part of the analysis, an understanding of the parameters of the word splitter was obtained. The Keras tokenizer was used to process the word-sorted data to obtain a matrix of training, stable, and test datasets, as well as a dictionary of the frequency and number of words corresponding to the occurrences. The dimensionality of the data processed by the sorter was 20,000 × 100 for the training set, 8000 × 100 for the stable set, and 2000 × 100 for the test set, which accounted for 66.7%, 26.7%, and 6.7% of the dataset, respectively.

The ChnSentiCorp_htl_all dataset was a dataset compiled by Mr. Songbo Tan with 7766 hotel reviews, including 5322 positive reviews and 2444 negative reviews. The allocation for the dataset was 4660 training samples, 1553 validation samples, and 1553 test samples for various sentiment analysis-related experiments. They accounted for 60%, 20%, and 20% of the dataset, respectively. The labeling methods and some of the comment data are shown in Tables 3 and 4.

Table 3. Description of the ChnSentiCorp_htl_all dataset.

Field	Description
label	1 indicates a positive comment, 0 indicates a negative comment
review	Content

Table 4. Selected data from the ChnSentiCorp_htl_all dataset.

Serial Number	Label	Review
5612	0	The room is unimaginably small, it is recommended that large people do not choose, the average sleeping feet can not be straight. The room is not more than 10 square feet, and the color TV is 14...
7321	0	Our family took the kids to the "May Day". The hotel is a great place to stay, but it seems to be wrong. 1. The hotel is in addition to...
3870	1	I went to the West Hill on Saturday to pick oranges and thought it would be a good hotel to stay at when I passed by...
4057	1	Convenient transportation is within walking distance to Fisherman's Wharf and Macau Ferry Terminal...
1452	1	It is a very nice hotel with a big bed and very comfortable. The hotel staff is very friendly.

4.2. Evaluation Indicators

This paper used accuracy, F1 score, Macro F1, and binary cross entropy loss function as evaluation metrics. Accuracy provided a clear judgment of the model's performance; F1 score was the summed average of accuracy and recall, which takes into account the accuracy and recall of the classification model; and Macro F1 was the average F1 score per category, providing an overview of the overall performance assessment. Below are the calculation formulas.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

$$\mathrm{Pr}ecision(P) = \frac{TP}{TP + FP} \tag{8}$$

$$\mathrm{Re}call(R) = \frac{TP}{TP + FN} \tag{9}$$

$$F_1 = 2 \times \frac{P \times R}{P + R} \tag{10}$$

$$\mathrm{H(p, q)} = -\sum \mathrm{p(x)log(q(x))} \tag{11}$$

where TP indicates the number of sentiment predictions that are positive and correct, and TN shows the number of sentiment predictions that are negative and correct. FP suggests the number of harmful category errors predicted as positive. FN indicates the number of positive category errors predicted as unfavorable. The loss function was calculated using Equation (11) where p and q represent the true distribution and the prediction, respectively.

4.3. Model Parameter Settings

The model parameters and their descriptions are shown in Table 5.

Table 5. Model parameter settings.

Name of Experimental Parameter	Parameter Values
Max Length of Sentences	100
Size of Word Vector	100
Batch Size	100
Window Size	3, 4, 5
Epochs	10
Dropout_rate	0.5
Optimizer	Adam
Learning_rate	0.001

4.4. Comparative Tests

To verify the validity of the hybrid neural network model, several classical models were selected for comparison experiments.

(1) TextCNN: Used for sentiment classification of text, it is a single basic convolutional neural network sentiment analysis method. In this paper, it was optimized by layer stacking.

(2) LSTM: Used for sentiment classification of text, it is a single basic long- and short-term memory neural network sentiment analysis method. In this paper, the LSTM was enhanced by increasing its number and complexity.

(3) TextCNN-LSTM: The text data are first transformed into word vectors through the embedding layer, and then features at different levels are extracted through multiple convolutional kernels in the TextCNN part. These extracted features are then

transformed into a time series and handed over to the LSTM part for subsequent processing.

(4) BiLSTM-ATT: First, the text sequence is transformed into a word vector through the embedding layer. Next, an attention mechanism is introduced for weighting the contribution of different words to the output of a given input text sequence to obtain more accurate and important information.

(5) Attention-Based Convolutional Neural Network (ABCNN): Combining the attention mechanism and CNN to sentence modeling, the goal is to construct a new sentence model containing sentence contextual relationships by taking into account the correlations between sentences through the attention mechanism.

(6) BERT-ETextCNN-ELSTM: First, the input text sentences are processed by the BERT pre-training model to obtain the corresponding word vector representation. Then, the TextCNN is optimally fused with an LSTM enhanced by increasing the number and complexity through layer stacking into an ETextCNN-ELSTM, after which the obtained word vectors are input into the ETextCNN-ELSTM to capture the features in the text sequence to different degrees through multiple convolutional kernels.

4.5. Analysis of Experimental Results

The error and accuracy obtained by the BERT-ETextCNN-ELSTM model trained on the simplifyweibo_4_moods and ChnSentiCorp_htl_all datasets at different numbers of iterations are shown in Figures 8 and 9. We can see that the accuracy of the model on the training set reached its highest at the 10th iteration, and therefore the number of iterations for this model was chosen to be 10.

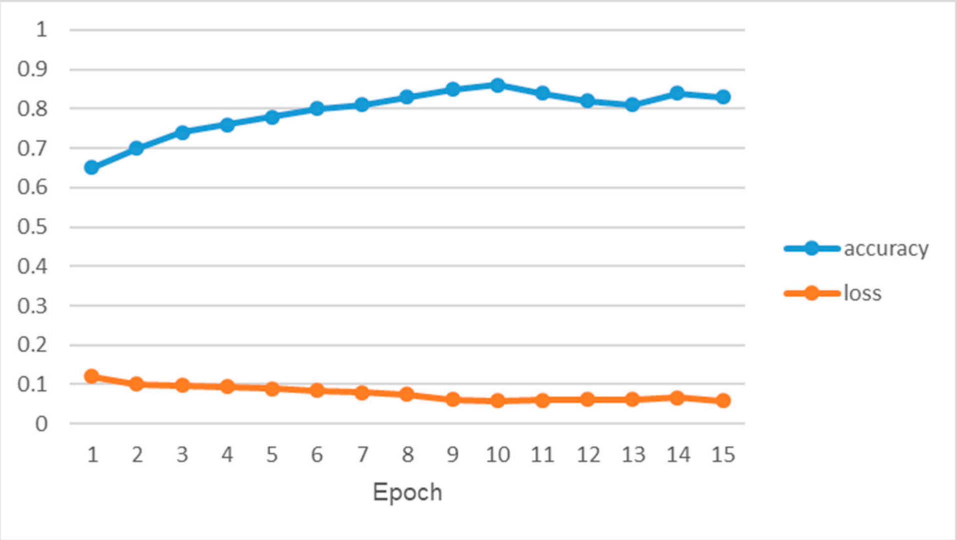

Figure 8. Loss and accuracy of the simplifyweibo_4_moods dataset.

From the above experiments, we can see that the number of iterations also affected the performance of the models, so we compared the results of each comparison model at different iterations to select the most appropriate number of iterations. Figures 10 and 11 show the experimental results for the six comparison models on the simplifyweibo_4_moods and ChnSentiCorp_htl_all datasets at different numbers of iterations.

From the above results, it can be seen that the BERT-ETextCNN-ELSTM model achieved the best sentiment analysis performance on both datasets compared to the other five comparison models, and it can also be seen that the best results were achieved when the number of iterations was 10, so the number of iterations for the model in this paper was set to 10.

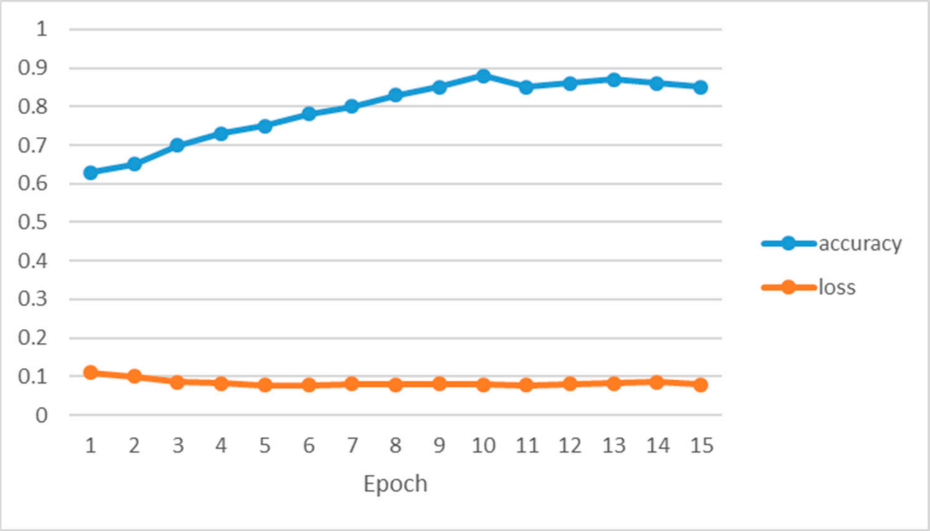

Figure 9. Loss and accuracy of the ChnSentiCorp_htl_all.

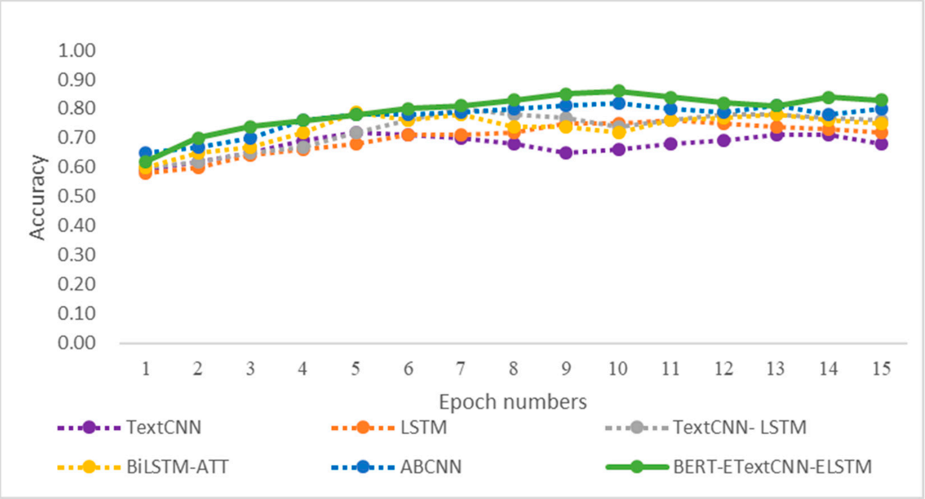

Figure 10. Accuracy of comparison models on the simplifyweibo_4_moods dataset.

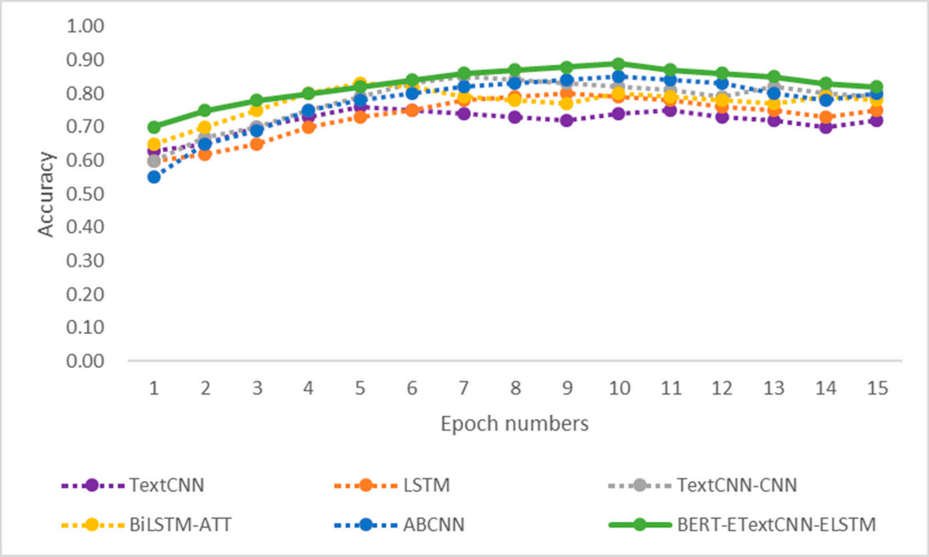

Figure 11. Accuracy of comparison models on the ChnSentiCorp_htl_all dataset.

In the training process of the model, this experiment introduced the dropout method. The dropout value is an important parameter, and a suitable value can make the model converge better, prevent the model from overfitting, and improve the performance of the model. Therefore, we chose different dropout values for training. The dropout values set in this experiment were [0.2, 0.3, 0.4, 0.5, 0.6, 0.7], and the best dropout value was selected from the training results of the model. The experiments were conducted on the simplifyweibo_4_moods dataset and the results of the experiments on the simplifyweibo_4_moods and ChnSentiCorp_htl_all dataset are shown in Figures 12 and 13. Through the results we can see that only the LSTM model worked best when the dropout value was 0.6, while the rest of the models achieved the best results when the dropout value was 0.5. The dropout value at this time could guarantee the accuracy of the results on the premise of the dropout value effectively preventing the model from overfitting, so the dropout value of the model in this paper was set to 0.5.

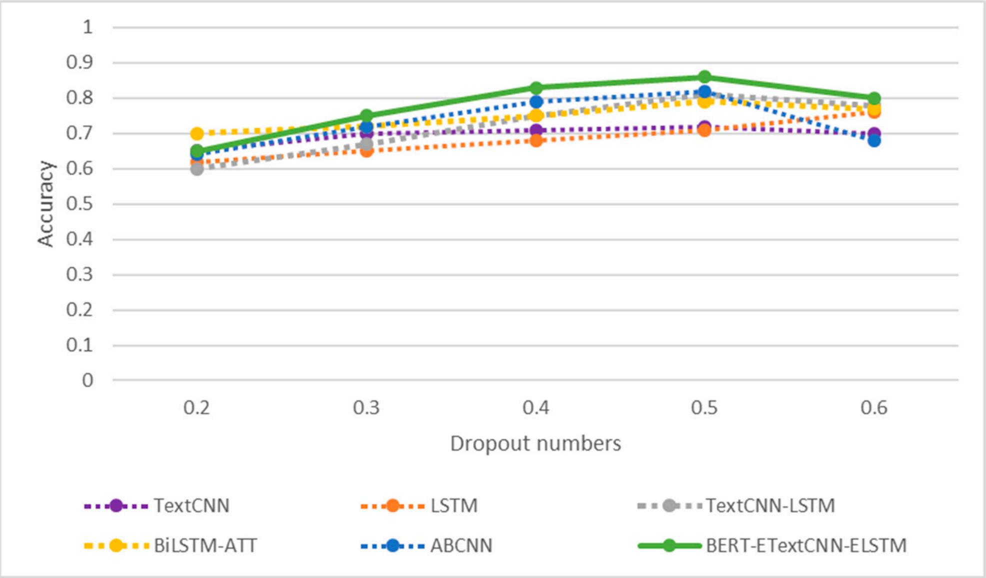

Figure 12. Accuracy of each model for the simplifyweibo_4_moods dataset for different dropout values.

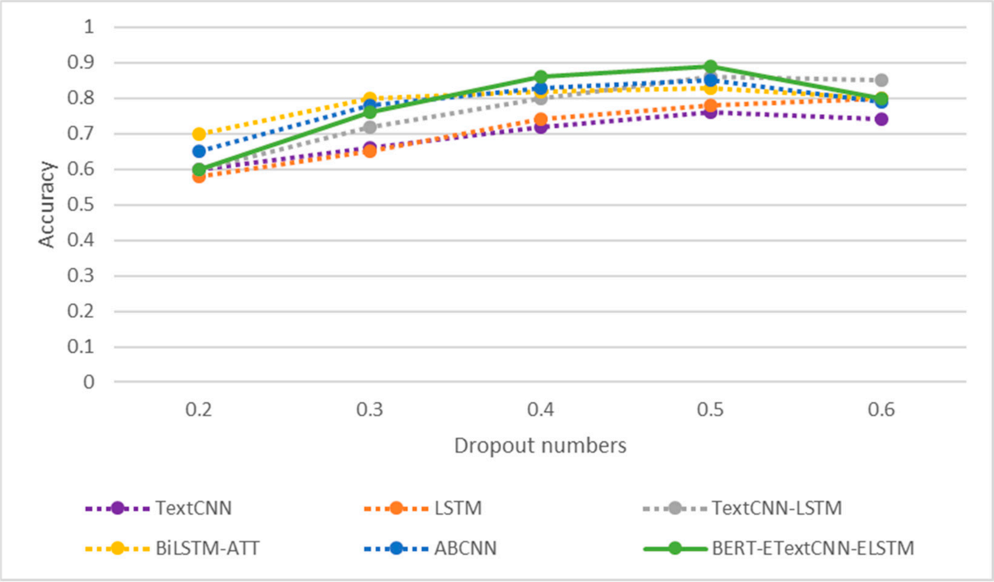

Figure 13. Accuracy of each model for the ChnSentiCorp_htl_all dataset for different dropout values.

In the process of gradient back propagation to update the parameters of the neural network, the optimizer used in this experiment was Adam. The Adam optimization algorithm is computationally efficient and converges quickly. To better exploit the efficiency of this algorithm, this paper chose different learning rate values to conduct experiments on the simplifyweibo_4_moods and ChnSentiCorp_htl_all datasets. The results on the simplifyweibo_4_moods and ChnSentiCorp_htl_all datasets are shown in Figures 14 and 15. From the experimental results, it can be seen that the model had the highest accuracy when the corresponding learning rate of Adam was 0.001. Therefore, the learning rate of the Adam optimizer in this paper was taken to be 0.001.

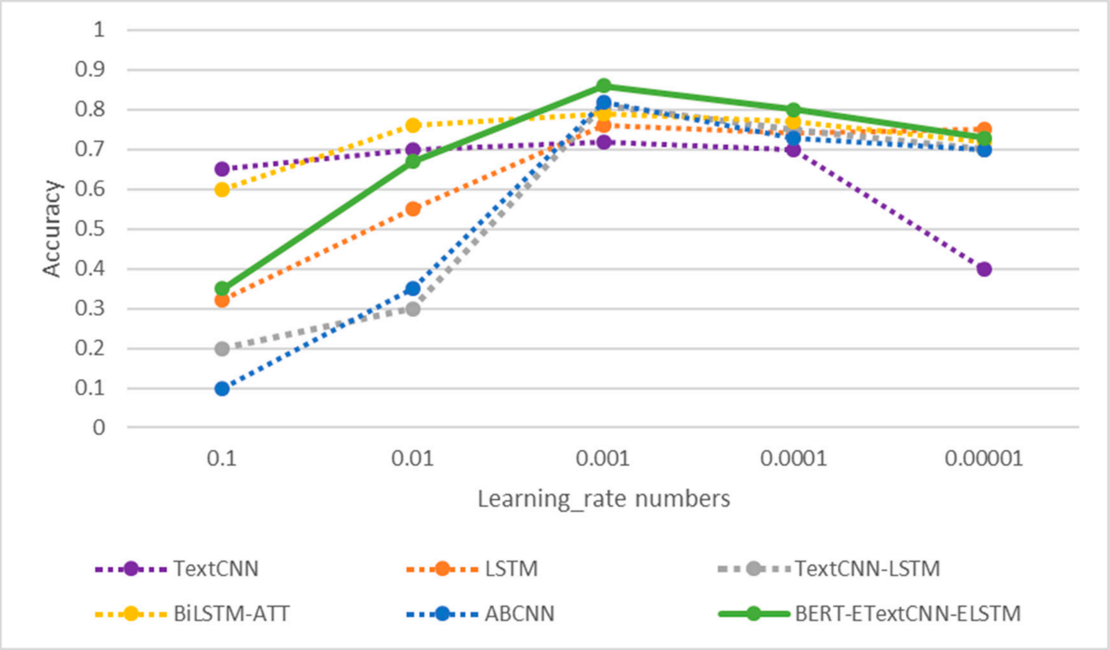

Figure 14. Accuracy of the simplifyweibo_4_moods dataset for each model at different learning rates.

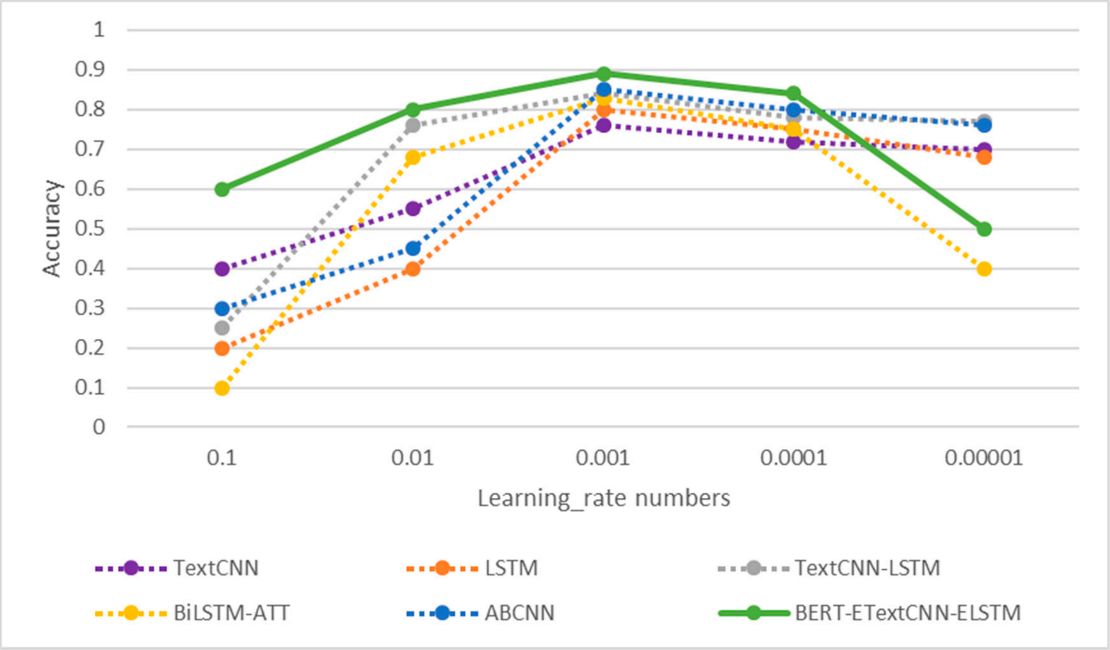

Figure 15. Accuracy of the ChnSentiCorp_htl_all dataset for each model at different learning rates.

The experimental results of the proposed model and other comparative models on the simplifyweibo_4_moods and ChnSentiCorp_htl_all datasets are shown in Tables 6 and 7 and Figure 16. To verify the effectiveness of the hybrid (combined, fused) neural network model proposed in this paper, using pre-trained models as well as optimized ones, several classical models were selected for comparison experiments. In the single neural network approach to sentiment analysis, the TextCNN and LSTM models were selected for comparison experiments. In the hybrid (combined, fused) neural network approach to sentiment analysis, among the sentiment analysis methods that introduce an attention mechanism, BiLSTM-ATT and Attention-Based Convolutional Neural Network (ABCNN) were chosen for comparison experiments. In both experiments, the best results of each model were selected for comparison.

Table 6. Table comparing experimental results on the simplifyweibo_4_moods dataset.

Model	Accuracy	F1 Score	Macro F1
TextCNN	0.72	0.71	0.69
LSTM	0.76	0.74	0.72
TextCNN-LSTM	0.81	0.79	0.78
BiLSTM-ATT	0.79	0.78	0.77
Attention-Based Convolutional Neural Network (ABCNN)	0.82	0.80	0.79
BERT-ETextCNN-ELSTM	0.86	0.85	0.84

Table 7. Table comparing experimental results on the ChnSentiCorp_htl_all dataset.

Model	Accuracy	F1 Score	Macro F1
TextCNN	0.76	0.75	0.74
LSTM	0.80	0.79	0.78
TextCNN-LSTM	0.86	0.85	0.83
BiLSTM-ATT	0.83	0.81	0.79
Attention-Based Convolutional Neural Network (ABCNN)	0.85	0.84	0.83
BERT-ETextCNN-ELSTM	0.89	0.88	0.86

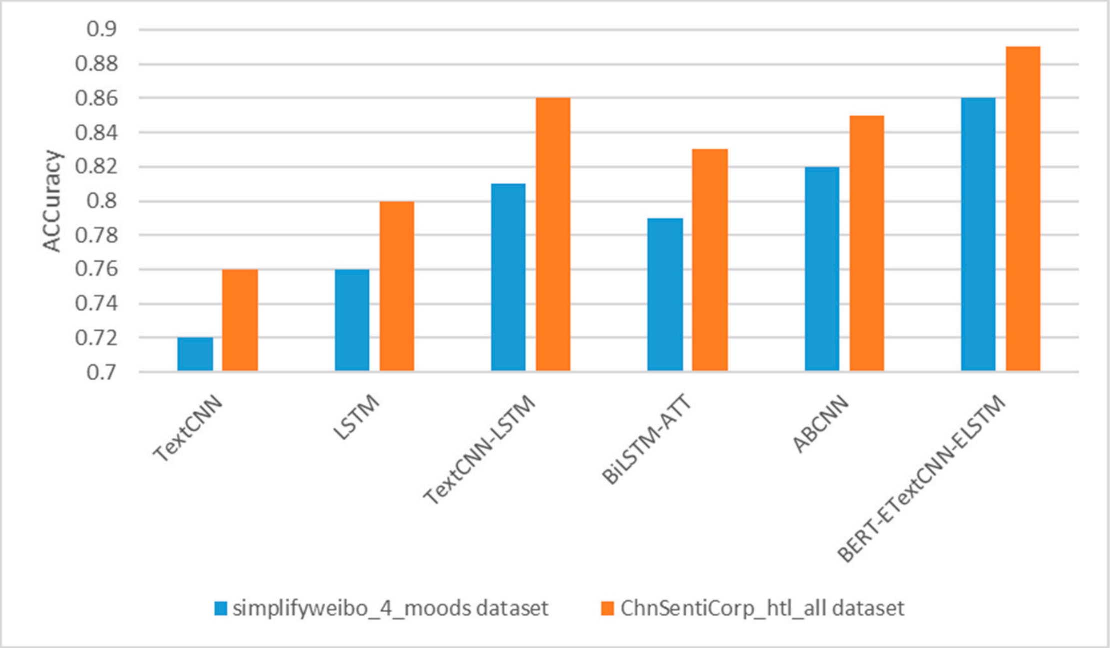

Figure 16. Accuracy of each model on different datasets.

From the experimental results, it can be seen that the BERT-ETextCNN-ELSTM model proposed in this paper achieved the best sentiment analysis performance on both the simplifyweibo_4_moods and ChnSentiCorp_htl_all datasets, with the highest accuracy, F1 value, and macro-average F1 value. From the results, it can be seen that the overall performances of TextCNN-LSTM, BiLSTM-ATT, Attention-Based Convolutional Neural Network (ABCNN), and the model in this paper, BERT-ETextCNN-ELSTM, were significantly higher than those of TextCNN and LSTM. Additionally, the hybrid (combined, fused) neural networks for sentiment analysis compared to single neural network approaches were studied, and the advantages of different approaches were considered before combining and improving these approaches. Their use for sentiment analysis achieved good results, indicating that this approach was significantly effective in alleviating the problem of reliance on the model's structure. Among the hybrid models, the performance of the model proposed in this paper, BERT-ETextCNN-ELSTM, was significantly higher than that of TextCNN-LSTM, BiLSTM-ATT, and ABCNN, indicating that the BERT model incorporated in this paper could better handle contextual information and deal with problems such as polysemy and ambiguity. In addition, the optimization of TextCNN-LSTM in this paper enabled the model to more fully exploit the deep semantic information of short textbooks, thus further improving sentiment analysis of comment data.

5. Conclusions

With the development of the Internet, comment data have become more diverse and the structure of comment data has become more complex. Traditional sentiment analysis methods are no longer able to produce results with great accuracy, and deep learning methods are constantly developing new models due to their ability to actively extract text features and their excellent performance in sentiment analysis tasks.

The research content of this paper aimed to address the shortcomings in deep learning and improve its sentiment analysis performance. The main contributions and findings of this thesis are as follows:

In response to the problem that traditional deep learning models cannot extract deep semantic information and that it becomes more difficult for traditional deep learning models to extract text features when the information from review data keeps changing, such as the emergence of new vocabulary, an optimized CNN-LSTM model was proposed to better complete the extraction of features. The model superimposed layers on the convolutional neural network, which not only reduced the complexity of the model and gradient disappearance due to redundancy of the model, but it also increased the output channels in the TextCNN network, enhanced the LSTM, increased the number and complexity of the LSTM, and achieved better extraction of data features.

By introducing the BERT model, our model could take full advantage of deep bidirectional contextual understanding to better capture the global semantic information of sentences. The pre-training capability of BERT and learning from a large corpus enabled our model to better understand Chinese text and perform an accurate analysis of sentiment. Experimental results on two publicly available datasets, simplifyweibo_4_moods and ChnSentiCorp_htl_all, validated the superiority of our model over current mainstream models and achieved better performance and results. This demonstrated the robustness and applicability of the model, as well as its effectiveness for Chinese sentiment analysis tasks. However, comment data from websites have complex issues such as imperfect expression and inaccuracy. This experiment will further refine the advancement of the algorithm since, for example, speech, images, and videos also intuitively express people's emotions, and the next work will also explore applications in the fields of speech, image, and video processing to improve the accuracy of the analysis.

Author Contributions: Conceptualization, L.D., T.Y. and Z.L.; Methodology, L.D. and T.Y.; Formal analysis, L.D., Z.L. and Q.G.; Software, L.D., T.Y., Q.G. and T.Y.; Validation, T.Y., Z.L. and Z.L.; Investigation, Z.L. and Q.G.; Resources, Q.G.; Writing-original draft, Q.G. All authors have read and agreed to the published version of the manuscript.

Funding: This paper was supported by the National Natural Science Foundation of China [Grant 61702462], the Henan Provincial Science and Technology Research Project [Grants 222102210010 and 222102210064], the Research and Practice Project of Higher Education Teaching Reform in Henan Province [Grants 2019SJGLX320 and 2019SJGLX020], the Undergraduate Universities Smart Teaching Special Research Project of Henan Province [Grant Jiao Gao (2021) No. 489-29], and the Academic Degrees & Graduate Education Reform Project of Henan Province [Grant 2021SJGLX115Y].

Data Availability Statement: The data presented in this study are openly available in https://datafountain.cn/datasets/54, (accessed on 30 August 2022).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wang, X.; Wei, F.; Liu, X.; Zhou, M.; Zhang, M. Topic sentiment analysis in twitter: A graph-based hashtag sentiment classification approach. In Proceedings of the 20th ACM International Conference on Information and Knowledge Management, Glasgow, UK, 24–28 October 2011; pp. 1031–1040.
2. Brauwers, G.; Frasincar, F. A survey on aspect-based sentiment classification. *ACM Comput. Surv.* **2022**, *55*, 1–37. [CrossRef]
3. Jia, K. Sentiment classification of microblog: A framework based on BERT and CNN with attention mechanism. *Comput. Electr. Eng.* **2022**, *101*, 108032. [CrossRef]
4. Sun, B.; Tian, F.; Liang, L. Tibetan micro-blog sentiment analysis based on mixed deep learning. In Proceedings of the 2018 International Conference on Audio, Language and Image Processing (ICALIP), Shanghai, China, 16–17 July 2018; pp. 109–112.
5. Jia, K.; Li, Z. Chinese micro-blog sentiment classification based on emotion dictionary and semantic rules. In Proceedings of the 2020 International Conference on Computer Information and Big Data Applications (CIBDA), Guiyang, China, 17–19 April 2020; pp. 309–312.
6. Hong, X.; Shaohua, X. Analysis on Web Public Opinion Orientation Based on Syntactic Parsing and Emotional Dictionary. *J. Chin. Comput. Syst.* **2014**, *35*, 811–813.
7. García-Méndez, S.; de Arriba-Pérez, F.; Barros-Vila, A.; González-Castaño, F.J. Targeted aspect-based emotion analysis to detect opportunities and precaution in financial Twitter messages. *Expert Syst. Appl.* **2023**, *218*, 119611. [CrossRef]
8. Fang, L.; Shao, D. Application of long short-term memory (LSTM) on the prediction of rainfall-runoff in karst area. *Front. Phys.* **2022**, *9*, 790687. [CrossRef]
9. Sangeetha, K.; Prabha, D. Sentiment analysis of student feedback using multi-head attention fusion model of word and context embedding for LSTM. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 4117–4126. [CrossRef]
10. Yadav, R.K.; Jiao, L.; Goodwin, M.; Granmo, O.C. Positionless aspect based sentiment analysis using attention mechanism. *Knowl. Based Syst.* **2021**, *226*, 107136. [CrossRef]
11. Li, Y.; Zhang, K.; Wang, J.; Gao, X. A cognitive brain model for multimodal sentiment analysis based on attention neural networks. *Neurocomputing* **2021**, *430*, 159–173. [CrossRef]
12. Banupriya, R.; Kannan, R. A convolutional neural network based feature extractor with discriminant feature score for effective medical image classification. *NeuroQuantology* **2020**, *18*, 1.
13. Febrian, R.; Halim, B.M.; Christina, M.; Ramdhan, D.; Chowanda, A. Facial expression recognition using bidirectional LSTM—CNN. *Procedia Comput. Sci.* **2023**, *216*, 39–47. [CrossRef]
14. Satrya, W.F.; Aprilliyani, R.; Yossy, E.H. Sentiment analysis of Indonesian police chief using multi-level ensemble model. *Procedia Comput. Sci.* **2023**, *216*, 620–629. [CrossRef]
15. Kale, A.S.; Pandya, V.; Di Troia, F.; Stamp, M. Malware classification with Word2Vec, HMM2Vec, BERT, and ELMo. *J. Comput. Virol. Hacking Tech.* **2022**, *19*, 1–16. [CrossRef]
16. Zheng, Y.; Zhang, R.; Wang, S.; Mensah, S.; Mao, Y. Anchored model transfer and soft instance transfer for cross-task cross-domain learning: A study through aspect-level sentiment classification. In Proceedings of the Web Conference 2020, Taipei, Taiwan, 20–24 April 2020; pp. 2754–2760.
17. Sun, L.; Lian, Z.; Tao, J.; Liu, B.; Niu, M. Multi-modal continuous dimensional emotion recognition using recurrent neural network and self-attention mechanism. In Proceedings of the 1st International on Multimodal Sentiment Analysis in Real-life Media Challenge and Workshop, Seattle, WA, USA, 16 October 2020; pp. 27–34.
18. Alhagry, S.; Fahmy, A.A.; El-Khoribi, R.A. Emotion recognition based on EEG using LSTM recurrent neural network. *Int. J. Adv. Comput. Sci. Appl.* **2017**, *8*, 355–358. [CrossRef]
19. Chen, Y. Convolutional Neural Network for Sentence Classification. Master's Thesis, University of Waterloo, Waterloo, ON, Canada, 2015.
20. Zhuge, Q.; Xu, L.; Zhang, G. LSTM Neural Network with Emotional Analysis for prediction of stock price. *Eng. Lett.* **2017**, *25*, 167–175.
21. Zhou, Q.; Wu, H. NLP at IEST 2018: BiLSTM-attention and LSTM-attention via soft voting in emotion classification. In Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, Brussels, Belgium, 31 October 2018; pp. 189–194.
22. Cheng, Y.; Sun, H.; Chen, H.; Meng, L.; Cai, Y.; Cai, Z.; Huang, J. Sentiment analysis using multi-head attention capsules with multi-channel CNN and bidirectional GRU. *IEEE Access* **2021**, *9*, 60383–60395. [CrossRef]

23. Munikar, M.; Shakya, S.; Shrestha, A. Fine-grained sentiment classification using BERT. In *2019 Artificial Intelligence for Transforming Business and Society (AITB)*; IEEE: Piscataway, NJ, USA, 2019; Volume 1, pp. 1–5.
24. Yu, L.; Chen, L.; Dong, J.; Li, M.; Liu, L.; Zhao, B.; Zhang, C. Detecting malicious web requests using an enhanced textcnn. In Proceedings of the 2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC), Madrid, Spain, 13–17 July 2020; pp. 768–777.
25. Bengio, Y.; Ducharme, R.; Vincent, P. A neural probabilistic language model. *Adv. Neural Inf. Process. Syst.* **2000**, *13*, 1137–1155.
26. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. In Proceedings of the Advances in Neural Information Processing Systems 26 (NIPS 2013), Lake Tahoe, NV, USA, 5–10 December 2013; Volume 26.
27. Kumar, A.; nee Khemchandani, R.R. Self-attention enhanced recurrent neural networks for sentence classification. In Proceedings of the 2018 IEEE Symposium Series on Computational Intelligence (SSCI), Bengaluru, India, 18–21 November 2018; pp. 905–911.
28. Kota, V.R.; Munisamy, S.D. High accuracy offering attention mechanisms based deep learning approach using CNN/bi-LSTM for sentiment analysis. *Int. J. Intell. Comput. Cybern.* **2022**, *15*, 61–74. [CrossRef]
29. Sharma, A.K.; Chaurasia, S.; Srivastava, D.K. Sentimental short sentences classification by using CNN deep learning model with fine tuned Word2Vec. *Procedia Comput. Sci.* **2020**, *167*, 1139–1147. [CrossRef]
30. Ullah, F.; Chen, X.; Shah, S.B.H.; Mahfoudh, S.; Hassan, M.A.; Saeed, N. A Novel Approach for Emotion Detection and Sentiment Analysis for Low Resource Urdu Language Based on CNN-LSTM. *Electronics* **2022**, *11*, 4096. [CrossRef]
31. Liu, S.; Lee, I. Sequence encoding incorporated CNN model for Email document sentiment classification. *Appl. Soft Comput.* **2021**, *102*, 107104. [CrossRef]
32. Hui, L.; Yaqing, C. Fine-Grained Sentiment Analysis Based on Convolutional Neural Network. *Data Anal. Knowl. Discov.* **2019**, *3*, 95–103.
33. Sangeetha, J.; Kumaran, U. A hybrid optimization algorithm using BiLSTM structure for sentiment analysis. *Meas. Sens.* **2023**, *25*, 100619. [CrossRef]
34. Yin, W.; Schütze, H. Multichannel variable-size convolution for sentence classification. *arXiv* **2016**, arXiv:1603.04513.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.