# Addressing Sentiment Classification in Short Text Comments Using BERT and LSTM

He Li

Department of Computer Science and Technology, Wenzheng College of Soochow University, Soochow, P.R. China
box33@ldy.edu.rs

**Abstract.** The prevalence of short text comments in the comment sections of social media platforms accelerates the rate of information dissemination. The diversity and unpredictability of comment content can affect the sentiments of viewers and their judgment of topics and interfere with social media platforms' control over hot topics and optimization of user experience. Prioritizing the content of the comments section and understanding user sentiment tendencies are important tasks. This paper is primarily focused on efficiently classifying user comments based on their sentiment tendencies. This study used comments from the health section of Zhihu as the experimental dataset and annotated it for the labeling task. A model combining Bidirectional Encoder Representations from Transformers (BERT) and Long Short-Term Memory (LSTM) was selected for training, and the sentiment classification task was completed. The study discovered that various hyperparameter values and optimizers on the experimental dataset influenced the model's performance. The accuracy of text sentiment classification for the experimental data set reached 73%. By analyzing the experimental results, the model can effectively complete the task of classifying text in the comment area based on the sentimental state and obtain better classification performance after appropriate optimization steps.

**Keywords:** Natural Language Processing, Sentiment Classification, BERT, LSTM

## 1 Introduction

As practical application scenarios for the Internet continue to be enhanced, the widespread use of social media has made it convenient to disseminate information between people. More and more users are willing to share their views and opinions on a specific topic actively. On online platforms such as websites and blogs, there is a section known as the comment section under specific content, providing users with a platform for further discussion and feedback after viewing unified content. The comment section enhances the main content's dissemination effect and provides a more diverse way for users to interact and share. In today's social media environment, comments on section content published online have become important content that users who browse information pay attention to. Some users even attach more importance to the comment area


Y. Wang (ed.), *Proceedings of the 2024 2nd International Conference on Image, Algorithms and Artificial Intelligence (ICIAAI 2024)*, Advances in Computer Science Research 115,
https://doi.org/10.2991/978-94-6463-540-9_83

than the main content. Usually, content comments in the comment area often contain diverse sentiment tendencies of different users. These expressed sentiments inclined texts will also have varying degrees of impact on subsequent reading users. On the other hand, the content discussed by each user may be completely unrelated to the topic, and the comment text is written in a casual and informal writing style [1]. This part of the content can significantly affect other users' access to information in the comment section and also lead to the platform incorrectly assessing the actual level of discussion on the topic due to the generation of a large amount of unrelated content, thereby interfering with the analysis of the actual content discussion.

Therefore, paying attention to the discussion of specific topics by users in the comment area can also better understand the importance and attitude of current users towards specific topics and content. Research by G Mishne et al. has proven that a substantial quantity of blog comments can well indicate the importance of blogs [2]. Understanding changes and trends in public sentiment can help content creators better understand user feedback and needs and further improve the quality of content. At the same time, for online social media platforms, it can provide a clearer understanding of user-profiles and hot trends and improve the overall platform user experience while ensuring customer stickiness and activity. If a substantial quantity of users express extremely negative sentiments towards a specific issue, timely control and guidance of the discussion content or theme in a positive direction can be achieved. In this case, it is important to efficiently obtain the sentiment tendencies of user comments and perform automated classification, which is a crucial area for research.

In the field of Natural Language Processing (NLP), predicting and categorizing user review text content is a crucial research challenge. NLP is an interdisciplinary subject that combines computer science and linguistics. Its research goals include text understanding, language generation, information retrieval and classification, machine translation, and more. In the early stages, NLP research mainly relied on sentiment analysis methods based on emotional dictionaries or machine learning to conduct research on classification tasks [3]. Neural networks started showing remarkable results in NLP with the advent of deep learning. During this period, model architectures for processing language text gradually emerged and were put into practical research work.

In related research, the development of large models for text classification tasks has gone through multiple stages, and many classic model architectures have emerged. In addition to basic models such as Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) models [4], Salur et al. proposed an M-Hybrid model [5], which achieved a classification success rate of 82.14% in the same data set, and its performance was higher than other basic models. Basiri et al. proposed the Attention-based Bidirectional CNN-RNN Deep Model (ABCDM) [6], the experiments have shown that this model can effectively detect sentiment polarity and complete classification tasks for long reviews and short tweet. Recently, models based on the Transformer architecture [7] have emerged, and associated studies have fully validated these models' effectiveness and versatility in NLP research [8,9].

Based on the above content, considering the characteristics of research automation in the field of NLP and the advantages of processing unstructured data, this study will use deep learning as the technical means, to propose and design a classification model based on the combination of Bidirectional Encoder Representations from Transformers (BERT) and Long Short-Term Memory (LSTM). The model's performance will be evaluated based on the final classification accuracy, aiming to achieve precise sentiment analysis and classification of user short-text comments.

## 2 Data and Method

### 2.1 Data Source

To achieve the task of sentiment classification of user comment texts in social media comment areas, this experiment chose to go to the health theme section of the Zhihu platform to obtain user comments. Zhihu is a diversified Chinese question-and-answer community website that provides users a platform to communicate and express their opinions in areas they are good at or interested in. As one of the most popular question-and-answer platforms on the Chinese internet, user comments and posts on Zhihu possess a certain degree of diversity and timeliness. Further analysis of these comments and posts can efficiently reveal users' attitudes and concerns towards specific topics.

This study randomly selects and crawls user comments from a portion of the comment section as the dataset and does not perform any additional processing on the text in the dataset. The final dataset contains 1307 texts. This study aims to study the model's accuracy for text sentiment classification, use supervised learning to train the model, and manually label the dataset. Since the research focuses on user comments on specific topics, discussions unrelated to the specific topic would interfere with the analysis of the actual content discussion. Therefore, in addition to introducing "positive," "neutral," and "negative" as three "sentiment" states as classification labels, an attempt is made to introduce the "discussion of disease types" label simultaneously to reduce interference.

The labeling process invites three labelers to explain the labeling standards before labeling. Two labelers then independently conducted the first round of labeling. The third member integrated the results from the first two labels and made corrections. Specific annotated information samples are shown in Table 1. The labels for "sentiment" and "discussion of disease types" are shown in Table 1's first and second columns, respectively. The third column indicates the corresponding text content for each label, with the bolded text showing words related to the text's label classification. The text in the third column has been translated, and the original content is in Chinese.

**Table 1.** Examples of experimental data set labels and text content

| Sentiment | Discussion of disease types | Text |
|---|---|---|
| Positive | Traditional Chinese Medicine Healthcare. | **Maintaining good health** is indeed very important [Like] In fact, you can also try Chinese medicine to regulate your health **[Happy]** |
| Neutral | Epidemics and Influenza | To prevent **influenza A**, in addition to vaccination and enhancing immunity, effective protection must be carried out daily by wearing masks, reducing going out, and avoiding contact with crowds. |
| Negative | Mental Health | I used to be seriously **depressed** especially want to **die**, slit my wrists scars have not disappeared after ten years or so, slit my wrists is not a light hand and still did not die, messy swallowing pills also did not die. |
| Neutral | Others | Since the 1990s, schools have only focused on grades and admission rates, leading to today's situation. |
| Positive | Mental Health | My **anxiety** is finally out of the way, and I can now go to work **without worrying** about it...you guys do the same! |
| Negative | Others | I feel like it will be **useless** if this continues, and I don't know how to solve it. |

## 2.2 BERT

In this study, considering the versatility and practical feasibility of language pre-trained models, the selection of the BRET model matches more closely with the requirements of the project implementation stage.

2018 the Google team [10] proposed the Bidirectional Encoder Representations from Transformers (BERT) model. This model utilizes a bidirectional Transformer Encoder for feature extraction, coupled with the Masked Language Model method. It combines the advantages of previous language models and makes appropriate improvements, which have been widely utilized in the field of NLP. It can predict missing words in text by learning from a large text corpus, thereby training a universal language representation. Compared to traditional language models, BERT takes into account the current word's context as well as the impact of the sentence's other words on the current word. This allows BERT to comprehend the text's semantics more fully and is more suitable for handling downstream tasks such as sentiment classification. Fig. 1 shows the BERT model structure.

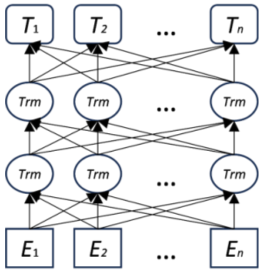


**Fig. 1.** BERT model structure

The abbreviation "Trm" in Fig.1 stands for Transformer. The Multi-Head Self-Attention module and the Feed Forward module are the two main modules in the Transformer Encoder module. After the data passes through these two modules, a Residual Connection is performed after each module to fit the training data and perform Layer Normalization. Fig. 2 shows the Transformer Encoder module's structure.

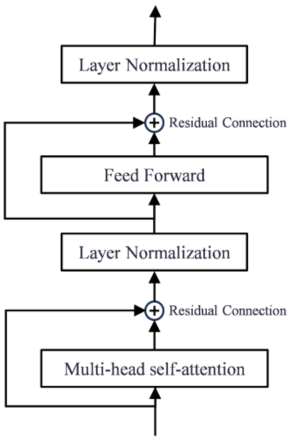


**Fig. 2.** Transformer Encoder module structure

## 2.3 LSTM

In the sentiment classification task, because the sentimental meaning of the text is often influenced by the context, to utilize contextual information better and achieve better model performance, this study chose to introduce an LSTM layer after the BERT model. This further processes the results obtained from BERT training to achieve higher model accuracy.

Long Short-Term Memory (LSTM) is a special neural network structure [11], its unique structure and excellent performance make it widely used in NLP research. Each neuron in LSTM contains a memory cell and three gating structures: the input gate, the output gate, and the forget gate. The input information flow is recorded and stored in the memory cell, while the gating mechanisms can more effectively capture long-distance dependencies in the information flow to retain important features and ensure the functionality and efficiency of information propagation. Compared to the basic models of traditional RNNs, the gating structures of LSTM can better alleviate issues such as gradient vanishing that may occur in basic RNN models.

### 2.4 BERT+LSTM Model

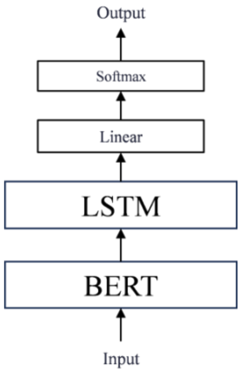


**Fig. 3.** BERT+LSTM model structure diagram

Fig.3 shows the structure of BERT+LSTM model. First, high-dimensional vector representations of the input semantic information are obtained by encoding the text data input using the BERT model. Subsequently, these representations are fed into the LSTM model as input vectors to capture their understanding of the text's sentiment information. Then, a linear classifier in the fully connected layer receives the LSTM model's output for classification. The final sentiment classification results are obtained by normalizing the outputs using the Softmax function. The Softmax function is defined as,

$$Softmax(x_i) = \frac{exp(x_i)}{\sum_{j=1}^{N} exp(x_j)} \quad (1)$$

where $x$ is the probability distribution vector obtained from the model's linear layer outputs, $x_i$ represents the $i$ element in the vector, and $N$ denotes the length of the $x$ vector.

During the model training process, the loss function is the function that is used to measure the difference between the model's predicted output and its true value. At the same time, a suitable loss function can strengthen the model's robustness and enhance its ability to classify data, reducing the negative effects of anomalous data on training outcomes. For example, Focal Loss, as proposed in the literature [12], attempts to increase the weight of the loss function on a small number of text categories to make the model focus more on learning a small number of samples to improve classification accuracy.

In the text classification task of this study, Cross-Entropy Loss [13] was selected as the loss function in the model training process. The loss function results obtained for each batch were used to update the model parameters, reducing the prediction error during training. The definition of the Cross-Entropy loss function is,

$$L_{label} = -\sum_{C=1}^{C} y_c \, log(\hat{x}_c) \tag{2}$$

where $C$ is the number of classes, $y_c$ is the distribution of the samples' actual labels, and $\hat{x}_c$ represents the model output vector's anticipated probability distribution following Softmax normalization.

Two distinct sets of labels in the text are One-hot encoded to enhance the model's generalization capability. In this study, losses for the "sentiment" label and the "discussion of disease types" label are calculated separately and then summed to obtain the final loss. This final loss is used as the new parameters for parameter updating in the next training batch.

### 2.5 Environment and Parameter Settings

The system environment used in this experiment is Ubuntu 20.04, with PyTorch 1.11.0 version as the deep learning framework, and the GPU used is RTX 3090 24GB. Considering efficiency and experimental environment factors, the BERT+LSTM model's parameters in this study are set as indicated in Table 2, "batch_size" refers to the number of samples used in each batch when model training, "epochs" indicates the number of times the entire dataset is trained, "max input length" indicates the longest text allowed in the model input, and "num_layers" signifies the number of stacked layers in the LSTM network.

**Table 2.** Parameter Settings

| Parameter | Values |
|---|---|
| batch_size | 9 |
| epochs | 1 |
| max input length | 512 |
| num_layers | 2 |

## 3 Experimental Results Analysis

### 3.1 Evaluation Metrics

The work of this experiment is to obtain the sentiment classification of the text through the model and judge the model's performance through the existing labels. Accuracy is the evaluation metric, showing how many samples were correctly predicted out of all the samples. Accuracy is defined as,

$$Accuracy = \frac{\sum_{i=1}^{N}(y_i = t_i)}{N} \quad (3)$$

where $y_i$ is the label prediction result of the $i$ sample obtained by the model, $t_i$ is the actual label of the $i$ sample, and $N$ is the number of samples. The total number of accurately predicted samples divided by the total number of samples $N$ yields the final accuracy. After determining the model evaluation indicators, try to adjust the parameters of the model and analyze the different parameter adjustment results.

## 3.2 The Impact of Optimizer on Model Performance

Considering that the optimizer is also a key factor affecting the model, the experiment's learning rate was kept the same at 5e-8, and different optimizers were selected to analyze the model performance. Run and train the model ten times, and record the accuracy results of each training, with the final accuracy result being determined by taking the average of the 10 values. Table 3 shows the final results.

**Table 3.** Model Classification Accuracy under Different Optimizers

| Optimizer | Accuracy |
|---|---|
| AdamW | 0.7115 |
| Adam | 0.6009 |
| Adamax | 0.6867 |
| Radam | 0.5185 |

The accuracy results in Table 3 show that when the learning rate in the optimizer is set at a fixed value, the model achieves the highest classification accuracy of 0.7115 when using the AdamW optimizer. However, when choosing Radam as the optimizer, the model's classification accuracy is only 0.5185, which is lower than when using the other three optimizers. Judging from the results, when training the same model, the different optimizers selected have their specific hyperparameters and settings. Choosing the right optimizer and setting the appropriate model parameters can help the model perform better.

## 3.3 The Impact of Learning Rate on Model Performance

Use Adam as the model optimizer, try to change the parameter setting value of the learning rate in the optimizer, and check the change in the model classification accuracy. Also run the model to train 10 times and record the results of each time, with the final accuracy result being determined by taking the average of the 10 values. Table 4 shows the classification accuracy results for different learning rates. Meanwhile, keep the rest of the settings unchanged, use Adam as the model optimizer, and choose to set different learning rates to experiment. Compare the probability distribution vector results obtained by the trained model after normalization by the Softmax function. Table

5 shows the vector results output after model training, and the data represents the model's predicted probabilities for the three categories [negative, neutral, positive]. Select the first sample vectors from batches 1, 15, 45, 75, 100, and 145 for recording and analysis.

**Table 4.** Model Classification Accuracy under Different Learning Rate

| Learning rate | Accuracy |
|---|---|
| 1e-6 | 0.7352 |
| 5e-7 | 0.7364 |
| 3e-7 | 0.7275 |
| 1e-7 | 0.7093 |
| 1e-8 | 0.7073 |
| 5e-8 | 0.6010 |
| 5e-10 | 0.5028 |

**Table 5.** Predicted Sample Vectors

| Learning rate / batch num | 3e-6 | 1e-7 | 5e-8 |
|---|---|---|---|
| 1 | [0.3192, 0.3454, 0.3354] | [0.3584, 0.3268, 0.3148] | [0.3584, 0.3268, 0.3148] |
| 15 | [0.2646, 0.4674, 0.2680] | [0.3520, 0.3297, 0.3182] | [0.3107, 0.3425, 0.3468] |
| 45 | [0.2079, 0.5668, 0.2254] | [0.3495, 0.3350, 0.3155] | [0.3165, 0.3395, 0.3440] |
| 75 | [0.1725, 0.6149, 0.2126] | [0.3440, 0.3456, 0.3104] | [0.3128, 0.3449, 0.3424] |
| 100 | [0.1304, 0.7029, 0.1666] | [0.3367, 0.3551, 0.3082] | [0.3092, 0.3547, 0.3360] |
| 145 | [0.0792, 0.8074, 0.1134] | [0.3279, 0.3682, 0.3039] | [0.2791, 0.4164, 0.3045] |

According to Table 4's experimental data, the Adam optimizer provides the model with higher classification performance when the learning rates are set to 5e-7 and 1e-6. This results in accuracies of 0.7364 and 0.7352, respectively. Meanwhile, observing the trend of accuracy changes, as the set learning rate value decreases, the accuracy also shows a corresponding decreasing trend. Only 0.5028 is the model's classification accuracy when the learning rate is set to 5e-10.

Table 5's experimental findings demonstrate that when the learning rate is adjusted, the model's predicted probability distribution tends to favor one of the elements in the classification, which remains consistent in subsequent training. For example, the output changes from [0.3192, 0.3454, 0.3354] in the first batch to [0.0792, 0.8074, 0.1134] in the 145th batch when the learning rate is set to 3e-6, with an increase in the predicted probability for the "neutral" class and a decrease in the predicted probability for the "positive" or "negative" classes.

The BERT+LSTM model was utilized for text classification tasks using the experimental dataset in this experiment. After optimization and parameter tuning, the model achieved a maximum accuracy of 73% for sentiment classification on the selected dataset.

The parameter setting of the model optimizer will affect its performance. Setting appropriate parameters for the learning rate can make the model perform better. If the learning rate is too large, the model may fail to effectively converge during the optimization process, leading to an inability to learn the necessary features and causing the predictive result to approach a fixed value. Conversely, if the learning rate of the model optimizer is set too low, in addition to causing the model training time to become longer, the model may incorrectly converge to local feature values during the training process, resulting in a decrease in accuracy.

## 4 Conclusion

This paper explores the importance of user comments in the comment section of social media platforms for user experience and staying informed about hot topics. The sentiment expressed in user comments texts is crucial and requires focused collection and analysis. Regarding sentiment classification tasks, the neural network-based language model in NLP is a reasonably well-developed and effective research solution. This study chose a model that combines BERT and LSTM for classification experiments. The study confirmed the effectiveness of this model in the task of sentiment judgment and classification of user short text comments in social media comment areas by analyzing the accuracy results obtained after the model was trained under various parameter changes. It has a certain reference significance for short text sentiment classification tasks.

In the process of research and experiments, when analyzing data set samples, the number of some specific labels is too high, which may cause the model training results to directly tend to the label with the most significant number after normalization. As for introducing two sets of labels, you can consider using one of the labels as an auxiliary task, using different models to process it separately, and adding a shared layer to enhance the connection between the two labels.

At the same time, for the classified label data set, only classifying the three categories of positive, negative, and neutral based on sentiment cannot fully cover the changes in sentiment when language is expressed. For example, there may be different levels of positive text expressions. Future experiments could consider treating sentiment labels as continuous dimensions.

## Reference


1. Wankhade, M., Rao, A. C. S., Kulkarni, C.: A survey on sentiment analysis methods, applications, and challenges. Artificial Intelligence Review 55(7), 5731-5780(2022)
2. Mishne, G., Glance, N.: Leave a reply: an analysis of weblog comments. Third annual workshop on the Weblogging ecosystem. (2006)
3. Wang, Y., Zhu, J., Wang, Z., BAI, F., Gong, J.: Review of applications of natural language processing in text sentiment analysis. Journal of Computer Applications 42(4), 1011(2022)
4. Yin, W., Kann, K., Yu, M., Schütze, H.: Comparative study of CNN and RNN for natural language processing. arXiv preprint arXiv:1702.01923(2017)
5. Salur, M. U., Aydin, I.: A novel hybrid deep learning model for sentiment classification. IEEE Access 8, 58080-58093(2020)
6. Basiri, M. E., Nemati, S., Abdar, M., Cambria, E., Acharya, U. R.: ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis. Future Generation Computer Systems 115, 279-294(2021)
7. Gillioz, A., Casas, J., Mugellini, E., Abou Khaled, O.: Overview of the transformer-based models for nlp tasks. In 15th Conference on Computer Science and Information Systems (FedCSIS), pp. 179-183. IEEE(2020)
8. Cui, Y., Che, W., Liu, T., Qin, B., Yang, Z.: Pre-training with whole word masking for chinese bert. IEEE/ACM Transactions on Audio, Speech, and Language Processing 29, 3504-3514(2021)
9. Müller, M., Salathé, M., Kummervold, P. E.: Covid-twitter-bert: A natural language processing model to analyse covid-19 content on twitter. Frontiers in Artificial Intelligence 6, 1023281(2023)
10. Devlin, J., Chang, M. W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805(2018)
11. Yang, S., Yu, X., Zhou, Y.: Lstm and gru neural network performance comparison study: Taking yelp review dataset as an example. In 2020 International workshop on electronic communication and artificial intelligence (IWECAI), pp. 98-101. IEEE(2020)
12. Lin, T. Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision, pp. 2980-2988(2017)
13. Mao, A., Mohri, M., & Zhong, Y.: Cross-entropy loss functions: Theoretical analysis and applications. In International Conference on Machine Learning , pp. 23803-23828. PMLR(2023)