*Electronics, Communications and Networks*
*A.J. Tallón-Ballesteros (Ed.)*


*doi:10.3233/FAIA241391*

# Research and Application of Text-Based Sentiment Analytics

Jiadong SUN[a,1], Mengnan WANG[b], Defen REN[a], Deji CHEN[a]
[a]*Wuxi University, Wuxi, Jiangsu, China*
[b]*Nanjing University of Information Science & Technology,Nanjing, Jiangsu, China*

**Abstract.** Text-based sentiment analytics is an important research direction in the field of natural language processing, and it is widely applied in identifying emotional tendencies. The proliferation of e-commerce platforms has underscored the criticality of emotional tendencies and user experiences delineated in product reviews for gauging product quality and user satisfaction. This study presents a developed text-based sentiment analysis system tailored to product reviews on e-commerce platforms. Three distinct text-based sentiment classification models based on Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and Bidirectional Encoder Representations from Transformers (BERT) were constructed and implemented. Subsequently, sentiment classification and comparative experiments were conducted utilizing real-world product review data. The experimental outcomes underscore that the BERT model outperforms others in terms of the Area Under the Curve (AUC) metric, yielding superior results in the domain of text sentiment analysis. This paper extensively deliberates on the fusion of deep learning methodologies into sentiment analytics, encapsulating the entire spectrum of data collection, preprocessing, and model analysis, with an overarching goal of providing a reference point for research endeavors in analogous realms.

**Keywords.** Deep Learning, BERT, Text Analysis, Sentiment Analysis

## 1. Introduction

With the rapid growth of e-commerce platforms, product reviews have become a critical factor in influencing consumer purchase decisions. Potential buyers often rely on reviews left by previous customers to evaluate whether a product meets their expectations before making a purchase. These reviews serve as direct feedback from real users and provide valuable insights into product quality, usability, and customer satisfaction. For merchants, product reviews offer a rich source of data to understand market trends, consumer preferences, and areas for improvement [1]. It is widely accepted that positive or negative reviews can significantly impact a product's sales performance [2].

Sentiment analysis, which aims to automatically detect and classify the emotions expressed in text, has become an essential tool for analyzing large-scale product reviews. At present, sentiment analysis methodologies mainly include two categories: dictionary-based, and machine learning-based. The dictionary-based sentiment analysis method uses a pre-built sentiment dictionary to calculate the sentiment tendency of the whole passage of texts through keyword matching. This approach is straightforward and user-

[1] Corresponding Author, Jiadong Sun, School of IoT Engineering,Wuxi University, No. 333, Xishan Avenue, Wuxi, Jiangsu, The China;E-mail:sunjiadong@cwxu.edu.cn.

friendly, but it has limitations in capturing the semantic context of the text, which may lead to potential ambiguity; furthermore, it cannot make detection when the keywords are not matched [3].

Sentiment analysis methods based on machine learning encompass traditional machine learning and deep learning approaches. The former has achieved significant accomplishments through the extraction of emotional features and the combination of classifiers. However, a notable limitation is its tendency to overlook contextual semantics. In contrast, the latter adopts hybrid neural networks, coupled with attention mechanisms, to effectively capture contextual semantic information, thus improving the accuracy of text sentiment analytics [4].

This paper proposes a deep learning-based sentiment analysis system for product reviews, leveraging RNN, LSTM, and BERT models. The models are trained on real-world e-commerce review data, and a comprehensive comparison is performed to evaluate their performance. By analyzing product reviews using these advanced models, this study aims to provide valuable insights into the application of deep learning in text-based sentiment analytics and contribute to the growing field of natural language processing (NLP).

## 2. Related theories and technologies

This section outlines the core neural network models used in this study—Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and Bidirectional Encoder Representations from Transformers (BERT)—and explains their relevance in performing text-based sentiment analysis on e-commerce product reviews.

### *2.1. Recurrent Neural Network (RNN)*

In sentiment analysis tasks, the order of text and contextual information are crucial, which is why Recurrent Neural Networks (RNNs) excel in this area. RNNs introduce a unique "memory" mechanism that effectively captures and retains temporal dependencies in input sequences. This capability allows RNNs to make informed judgments about the sentiment state at a given moment based on historical information from the input sequence.

The structure of an RNN consists of an input layer, hidden layer, and output layer. Unlike traditional feedforward neural networks, RNNs include recursive connections in the hidden layer that allow information to be passed between time steps [5]. As shown in Figure 1, at each time step, the RNN receives an input vector while also taking the hidden state from the previous time step as one of its inputs. Each hidden unit processes these inputs through an activation function (commonly using tanh or ReLU) and updates its state. Specifically, the update of the hidden state can be expressed as:

$$h_t = f(W * h_{t-1} + U * x_t + b_h) \tag{1}$$

$$y_t = g(V * h_t) \tag{2}$$

Where $h_t$ is the hidden state at the current time step, $h_{t-1}$ is the hidden state from the previous time step, $x_t$ is the current input, W,U and V are weight matrices, $b_h$ is the bias term, and f is the activation function. This recursive structure enables RNNs to maintain information flow within the sequence, capturing contextual information and maintaining consistency in sentiment judgments. In the output layer, the RNN converts the hidden states into sentiment classification results, typically using the softmax function for multiclass classification. Through this approach, RNNs can effectively handle emotional expressions in natural language, particularly when the context and order of sentiments are crucial for understanding.

RNNs have demonstrated exceptional performance in sentiment analysis applications. For example, Durga et al. used the BERT-large-cased model for sentiment classification on Twitter, restaurant, and laptop datasets, and evaluated the model's performance using a confusion matrix. The model was fine-tuned using Stochastic Gradient Descent (SGD) [6]. Additionally, Yang et Yang et al. used an RNN-based model for sentiment analysis, applying it to different datasets and validating the model's effectiveness through experiments [7]. Moreover, Mahajan and Chaudhary's applied an RNN-based model for sentiment analysis on Twitter data, using Google Translator to handle non-English text. Their experiments demonstrated improved performance compared to traditional algorithms like Maximum Entropy and Naive Bayes [8].

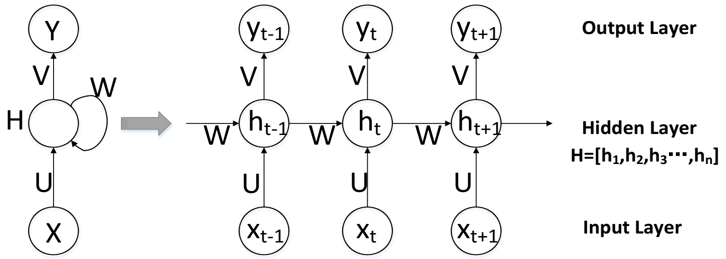


**Figure 1.** Schematic diagram of RNN model.

Despite the multitude of advantages of RNNs in the realm of sentiment analysis, they are susceptible to encountering the issue of gradient vanishing particularly in the handling of lengthy sequences. To address this issue, this paper subsequently introduces Long Short-Term Memory (LSTM) models, which are an improved version of RNNs capable of better capturing long-range contextual dependencies [9].

## *2.2. Long short-term memory network (LSTM)*

Long Short-Term Memory (LSTM) networks are a variant of Recurrent Neural Networks (RNNs) specifically designed to overcome common issues such as gradient vanishing and gradient explosion, which often hinder the training of deep networks. LSTMs excel at managing both long-term and short-term dependencies in sequential data, making them particularly effective for tasks involving time series and natural language processing [10]. The unique architecture of LSTM networks incorporates memory cells within the hidden layers, which facilitate the flow of information through three distinct gates: the input gate, the forget gate, and the output gate (Figure 2) [11].

The forget gate determines which information from the previous cell state should be discarded. It is calculated as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{3}$$

where $W_f$ is the weight matrix for the forget gate, $h_{t-1}$ is the hidden state from the previous time step, $x_t$ is the current input, $b_f$ is the bias term, and $\sigma$ is the sigmoid activation function.

The input gate decides which information from the current input should be added to the cell state, represented by:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{4}$$

Additionally, the candidate memory cell state is calculated as:

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (5)$$

where $W_i$ and $W_C$ are weight matrices for the input gate and candidate memory cell state, respectively, and $b_i$ and $b_C$ are the corresponding bias terms.

Next, the memory cell state $C_t$ is updated by combining the contributions from the previous cell state and the new candidate memory state:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (6)$$

The output gate determines what information to output based on the current memory cell state:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (7)$$

Finally, the hidden state is computed as:

$$h_t = o_t * \tanh(C_t) \quad (8)$$

This gating mechanism enables LSTMs to selectively store, retrieve, and discard information, thereby allowing the network to focus on the most pertinent data while filtering out noise. As a result, LSTMs can significantly enhance accuracy and stability in modeling complex sequences.

For instance, Venkataraman et al. utilized an LSTM-based model within a recurrent neural network (RNN) framework for sentiment analysis of Twitter emojis, classifying various sentiment categories like positive, neutral, and negative [12]. Additionally, Hu et al. employed a bidirectional LSTM model for sentiment classification, integrating multiple tags to enhance the semantic understanding of emotional texts. Their experiments showed a significant improvement in prediction accuracy for sentiment analysis [13].

Thanks to its robust memory mechanism, LSTM has become a powerful neural network suitable for a wide range of sequential data processing tasks, from language modeling to time-series forecasting.

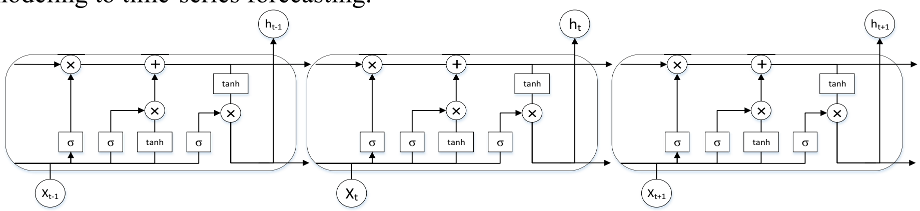


**Figure 2.** The chain structure of LSTM

## *2.3. BERT model*

BERT (Bidirectional Encoder Representations from Transformers) is a context-aware embedding model that leverages large amounts of unlabeled text for unsupervised pre-training, enabling it to learn rich linguistic representations [14]. When applied to downstream tasks, BERT requires only the addition of a task-specific output layer, followed by fine-tuning, to achieve high performance. One of its key strengths lies in its ability to understand sentence structure and capture complex contextual relationships, making it highly effective across various natural language processing (NLP) tasks. As shown in Figure 3, the core structure of BERT is based on the Transformer model, where each Transformer layer consists of multi-head self-attention mechanisms and feedforward neural networks [15]. The self-attention mechanism computes the relationships between each word in the input sequence using the following formula:

$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (9)$$

In this formula, Q represents the query vector, K is the key vector, V is the value vector, and dk is the dimension of the key vector. By calculating the relationships between each word and others, the model generates contextually relevant representations, allowing it to dynamically adjust the contribution of different words to the output. This capability enables BERT to capture both global and local dependencies when processing long texts [16,17].

The input text to BERT is first transformed into embedding representations, including word embeddings, positional embeddings, and segment embeddings. Word embeddings capture the semantic meaning of words, positional embeddings indicate the position of words in a sentence, and segment embeddings are used to differentiate between different input segments. These embeddings are then processed through multiple layers of Transformer encoders. The output from each layer contains not only the current layer's information but also retains information from previous layers, enhancing the model's contextual understanding.

Given the significant advantages of the BERT model in processing contextual text, many scholars have conducted research and applications on sentiment analysis using this model.For instance, Li et al. developed a Weibo text sentiment analysis model based on BERT [18]. Similarly, Sheng et al. proposed a BERT-based model for long Chinese text sentiment analysis, which effectively extracts core sentiments from long texts and significantly enhances sentiment analysis performance [19].The Transformer architecture in BERT, with its self-attention capability, is particularly suited for tasks requiring the capture of both global and local dependencies in text data, making it a dominant model in tasks such as machine translation, sentiment analysis, and question answering.

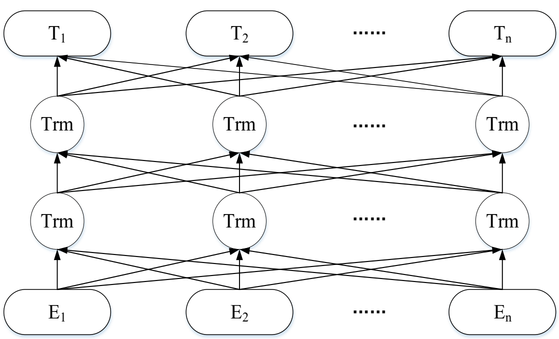


**Figure 3.** The structure model of BERT

## 3. Collection and preprocessing of data

### *3.1. Data collection*

In this experiment, crawler technology was implemented to collect real-world review data from the JD.com platform for the training data set. On the product interface of JD.com, when you click on a review, the web page will automatically request content from the server and analyze the results returned from the server. As can be seen, "productId" denotes the product number, and "page" denotes the page number of a product review. By changing different attributes in URL, JSON data can be retrieved

[20,21]. When crawling to collect the review contents, in order to avoid triggering the anti-crawler protection mechanism of JD.com server, a human operation will be simulated for random suspension of 2-4 seconds after one page of reviews is crawled, as shown in Figure 4.

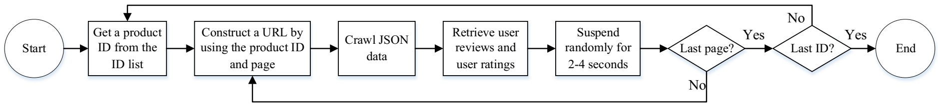


**Figure 4.** Data collection process

*3.2. Data preprocessing*

According to the common classification of e-commerce platforms, the user rating of 4-5 points is classified as a positive review, and 1-3 points as a negative review. Invalid information can often be found in reviews. This experiment removed the comments that contained only numbers or symbols and were less than 5 words in length. For repeated comments, the set function was used for de-duplication. Afterwards, the order of the data was shuffled to enhance randomness, and part of the data was deleted, so as to guarantee that there was the same quantity of positive and negative reviews. Finally, the preprocessed reviews were divided into two parts: the training set, and the test set. The training set accounts for 90%, and the test set accounts for 10%. The specific allocation is shown in Figure 5.

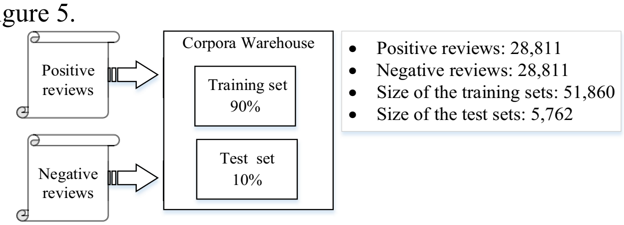


**Figure 5.** Allocation of samples

## 4. Experiment and analysis of models

*4.1. Experimental Environment*

The environment configuration for the sentiment analysis system of this study is illustrated in Table 1.

**Table 1.** The configuration of the training environment

| Operation environment | Name |
|---|---|
| Operating system | Linux |
| Graphics card | RTX 3090 |
| Programming language | Python |
| Deep learning framework | PyTorch |

In this study, the Area Under Curve (AUC) is primarily used as a key performance indicator for model evaluation. AUC measures the performance of a classification model by calculating the area under the Receiver Operating Characteristic (ROC) curve. The value of AUC ranges from 0 to 1, where 0.5 indicates that the model's predictive capability is equivalent to random guessing, and 1 indicates that the model has perfect predictive ability.

### 4.2. Segmentation of words and Construction of dictionary

In order to optimize the utilization of semantic information within the model, Chinese word segmentation is typically employed. In this experiment, the jieba word segmentation was used to facilitate Chinese word segmentation, and the optimal accurate mode was finally selected through comparative analysis. In addition, given that a large number of stop words would affect the models' training and execution speed, this experiment also filtered and deleted the word list established after word segmentation based on the "List of Chinese Stop Words". The resulting filtered dictionary consisted of 10,281 words, representing a decrease of 466 words compared to the pre-filtered version.

When the dictionary was constructed, the words were sorted in a descending order according to the frequency of their occurrences in the corpus. The words appearing less than 5 times were removed, in order to reduce the complexity of the model's subsequent computing. In addition, special marks were added to identify the unknown tokens; and special tokens were filled in sentences to unify the length of sentences. Afterwards, the words were encoded to obtain their corresponding encoding mappings. Finally, the constructed dictionary was used to transform all the corpora into coding vectors that can be processed by computer.

The figure 6 shows the distribution of all comment lengths in the corpus. As can be seen, the vast majority of sentences are within 300 characters, accounting for 98.6% of the total. To reduce the model's computational complexity, in subsequent processing, comments exceeding 300 characters will be directly truncated, while shorter ones will be padded with special symbols.

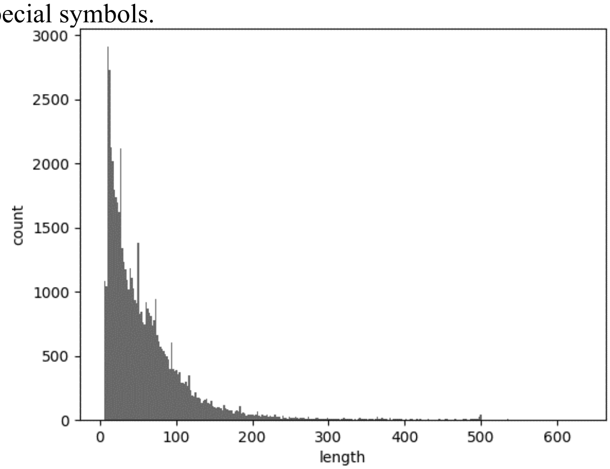


**Figure 6.** The distribution of corpus length.

### 4.3. Building and training of models

In the initial construction, the BERT-based model only added a fully connected layer and used the sigmoid activation function for processing. Although the AUC value of the training results was high, the generalization ability was relatively poor. To improve the model's generalization ability, mean_max_pool was subsequently employed.The structure of the pooling layer is shown in Figure 7.

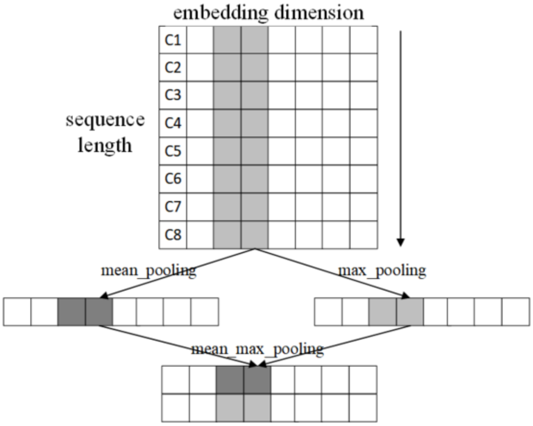


**Figure 7.** Pooling layer diagram.

In view of the feature that reviews on products are generally sequential texts, this experiment constructed several sentiment analytics models based on RNN, LSTM and BERT technologies for comparison and analysis. The process is shown in Figure 8.

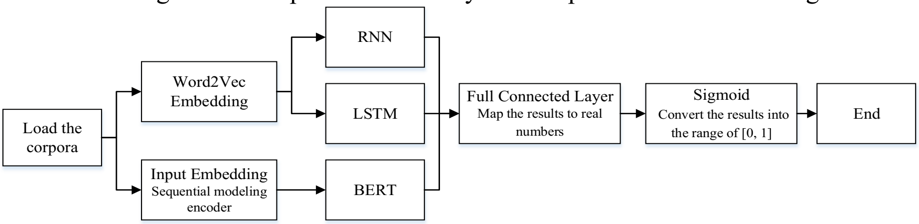


**Figure 8.** The process of creating the sentiment analytics models

1) The preprocessed corpora were loaded, and the sentences were normalized and tensorized to ensure uniform length. After word segmentation and removal of stop words, the maximum length of all comment words was 230. In order to reduce the computational complexity of the models, the length of the sentences was set to 200, whereby longer sections were truncated, and shorter sections were padded.

2) In order to improve the data quality, the RNN and LSTM models firstly used Word2vec as the first embedding layer for word vector representation; and the parameters of this layer were kept consistent during the training process. Then, separate construction was made based on different models. Finally, a fully connected layer was included; the results were mapped to real numbers; and the sigmoid activation function was used to convert the result range to [0, 1].

3) Training of RNN/LSTM models. This step involved the implementation of a combination and comparison experiment to enhance performance. This experiment tested scenarios with 1, 2, and 3 recurrent layers, and with 128, 256, and 512 hidden nodes in each layer. Ultimately, under the configuration with a learning rate (lr) of 0.01, embed_size of 100, and batch_size of 1024, the RNN model exhibited optimal performance when consisting of 2 RNN layers with 128 hidden nodes per layer. Conversely, the LSTM model performed best with 3 LSTM layers, each containing 512 hidden nodes. As shown in Figure 9, the RNN-based model iterated for 23 epochs, with the maximum AUC of 0.9345 appearing in the 19th round; and the LSTM-based model

iterated for 17 epochs, with the maximum AUC of 0.9445 appearing in the 13th round. The LSTM model experienced significant overfitting during the training process.

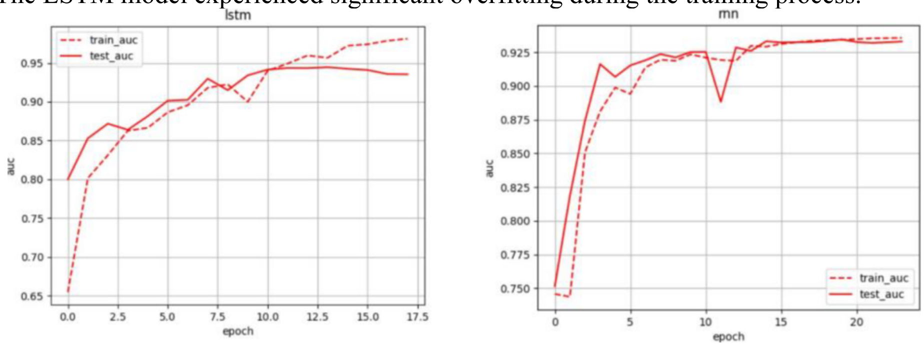


**Figure 9.** Parameter training diagram for RNN/LSTM-based models

4) Training of BERT model. The BERT model underwent training using a single additional fully connected layer with a sigmoid activation function initially. However, the resultant training outcomes exhibited notably high AUC values, albeit with limited generalization capability. To enhance the model's generalization capacity, the mean_max_pool adjustment was implemented for optimization and refinement. As depicted in Figure 10, the BERT model underwent a total of 138 epochs, with the highest AUC observed in the 133rd epoch, reaching a value of 0.9797.

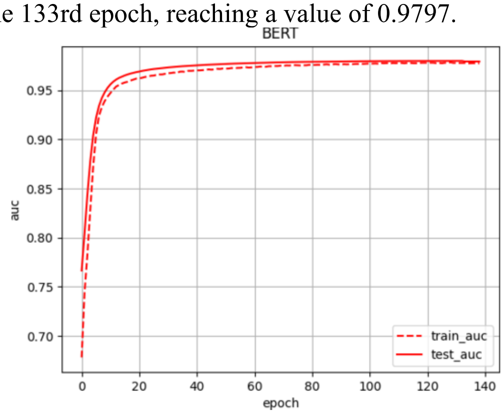


**Figure 10.** Parameter training graph for the BERT-based model

## *4.4. Result analysis and application*

This paper only adopted the AUC indicator for comparison and evaluation. Unlike other indicators, AUC provides a comprehensive analysis of the predictive accuracy of methods from a holistic perspective. Previous relevant research has also emphasized the selection of AUC as the preferred indicator, as evidenced by the work of scholar, such as Huang Danyang, Zou Lie, and Wu Yiteng [10]. The AUC value typically ranges between 0.5 and 1; with higher values signifying superior discriminative ability of a model. As shown in Table 2, all of the three models delivered very good results on the training data; and the AUC value of the BERT model reached the maximum, indicating that it can perform sentiment analysis more accurately.

**Table 2.** The training results of the three models

| Model | epoch | AUC value |
|---|---|---|
| RNN | 23 | 0.9345 |
| LSTM | 17 | 0.9445 |
| BERT | 138 | 0.9797 |

In the experiment, the BERT model with the best performance was selected for application demonstration, and the optimal threshold was set to 0.2. As shown in Figure 11, this model can be used for specific product evaluations.

```
感觉手机非常好，特别喜欢，要是发热再好点就更好了
Positive review,Output value:0.98
----------
拍照清晰系统流畅 变焦超级给力 不愧是拍照手机
美中不足的是电池虽大但掉电有些快
Positive review,Output value:0.69
----------
性价比之王，辨识度很高，屏幕分辨率很清晰，拍照很清楚，无线充电是最方便的，还送了一个手机壳
Positive review,Output value:0.98
----------
旗舰手机，不玩游戏，只拍照，性能溢出，不多说了，会买的都懂
Negative review,Output value:0.01
----------
```

**Figure 11.** Schematic diagram of the application of the models

## 5. Conclusion

This paper endeavors to explore the prevalent topic of text-based sentiment analysis. The web scraping techniques were adopted to gather review data from JD.com. Specifically, the collected data underwent several preprocessing procedures including de-duplication, data cleaning, and Chinese word segmentation. The review texts were then classified into positive and negative categories, and split into training and test sets in a 9:1 ratio. Sentiment analysis experiments were conducted using deep learning methods such as RNN, LSTM, and BERT. Through experimental comparison and result analysis, the effectiveness and excellence of the BERT method in the field of sentiment analytics were verified. This study furnishes valuable insights for the future exploration and implementation of sentiment analytics.

The significance of this study is underscored by the proposal of a BERT-based sentiment analysis framework that not only enhances the accuracy of sentiment classification but also provides a more practical solution for analyzing e-commerce reviews. This research establishes a theoretical foundation and practical reference for future applications in sentiment analysis on e-commerce platforms and other social media. Furthermore, based on the findings of this study, future research could delve into integrating different deep learning models to further improve the effectiveness and applicability of sentiment analysis.

## Acknowledgments

This work was supported by the Wuxi University Research Start-up Fund for Introduced Talents. The authors would like to express their sincere gratitude for this financial support, which was instrumental in the successful completion of this research.

## References

[1] THAKUR R.Customer engagement and online reviews[J]. Journal of Re-tailing and Consumer Services,2018,41(3): 48-59.

[2] LIU Cheng-yu,RAN Qi. Attitudinal Resources in Customers' Remarks in CSC and Their Impacts on Potential Customers' Purchase Decisions[J].Journal of University of Science and Technology Beijing(Social Sciences Edition), 2017, 33(06): 1-7.

[3] Wu C H, Chuang Z J, Lin Y C. Emotion recognition from text using semantic labels and separable mixture models[J]. ACM transactions on Asian language information processing (TALIP), 2006, 5(2): 165-183.

[4] Salehinejad H, Sankar S, Barfett J, et al. Recent advances in recurrent neural networks[J]. arXiv preprint arXiv:1801.01078, 2017.

[5] Rezk, N.M.; Purnaprajna, M.; Nordström, T.; Ul-Abdin, Z. Recurrent neural networks: An embedded computing perspective[J]. IEEE Access 2020, 8, 57967‑57996

[6] Durga P, Godavarthi D. Deep-Sentiment: An Effective Deep Sentiment Analysis Using a Decision-Based Recurrent Neural Network (D-RNN)[J]. IEEE Access, 2023,11,(108433-108447).

[7] Yang F, Du C, Huang L. Ensemble sentiment analysis method based on R-CNN and C-RNN with fusion gate[J]. International Journal of Computers Communications & Control, 2019, 14(2): 272-285.

[8] Mahajan D, Chaudhary D K. Sentiment analysis using RNN and Google translator[C].2018 8th international conference on cloud computing, data science & engineering (Confluence). IEEE, 2018: 798-802.

[9] Sepp Hochreiter, Jürgen Schmidhuber; Long Short-Term Memory[J]. Neural Comput 1997; 9 (8): 1735‑1780.

[10] Abotaleb, Mostafa and Dutta, Pushan Kumar. Optimizing long short-term memory networks for univariate time series forecasting: a comprehensive guide[J]. Hybrid Information Systems: Non-Linear Optimization Strategies with Artificial Intelligence, edited by Ramakant Bhardwaj, Pushan Kumar Dutta, Pethuru Raj, Abhishek Kumar, Kavita Saini, Alfonso González Briones and Mohammed K.A. Kaabar, Berlin, Boston: De Gruyter, 2024, pp. 427-442.

[11] Yu Xiaojian ,Liu Guopeng ,Liu Jianlin ,Xiao Weilin.Stock Index Prediction Based on LSTM Network and Text Sentiment Analysis[J/OL].Chinese Journal of Management Science2024,32(08):25-35.

[12] Venkataraman J, Mohandoss L. FBO ‐ RNN: Fuzzy butterfly optimization ‐ based RNN ‐ LSTM for extracting sentiments from Twitter Emoji database[J]. Concurrency and Computation: Practice and Experience, 2023, 35(12): e7683.

[13] Hu J, Kang X, Nishide S, et al. Text multi-label sentiment analysis based on Bi-LSTM[C].2019 IEEE 6th International Conference on Cloud Computing and Intelligence Systems (CCIS). IEEE, 2019: 16-20.

[14] Kasthuriarachchy, B.; Chetty, M.; Shatte, A.; Walls, D. From General Language Understanding to Noisy Text Comprehension[J]. Appl. Sci. 2021, 11, 7814.

[15] Zhang Z, Wu Y, Zhao H, et al. Semantics-aware BERT for language understanding[C].Proceedings of the AAAI conference on artificial intelligence. 2020, 34(05): 9628-9635.

[16] Liu Siqin ,Feng Xuruirui.Text Sentiment Analysis Based on BERT[J].Journal of Information Security Research,2020,6(03):220-227.

[17] DUAN Dandan,TANG Jiashan, WEN Yong,YUAN Kehai.Chinese Short Text Classification Algorithm Based on BERT Model[J].Computer Engineering,2021,47(01):79-86.

[18] Li H, Ma Y, Ma Z, et al. Weibo text sentiment analysis based on bert and deep learning[J]. Applied Sciences, 2021, 11(22): 10774.

[19] Sheng D, Yuan J. An efficient long Chinese text sentiment analysis method using BERT-based models with BiGRU[C].2021 ieee 24th international conference on computer supported cooperative work in design (cscwd). IEEE, 2021: 192-197.

[20] WU Shufan.Analysis on Affective Polarity of E-commerce Users' Comments Based on Deep Learning[J].Journal of Binzhou University,2022,38(04):87-91.

[21] Chen Zhuo ,Jiang Xixi ,Zhang Xiaojuan.Influence of Network Structure Changes on Co-word Network Link Prediction[J/OL].Data Analysis and Knowledge Discovery,2024,8(02):114-130.