Sentiment Analysis of Scenic Users' Comment Using FastText and LSTM Model

Lei Shang[1], Yijia Wang[2], Xinqi Dong[2], Peiyao Niu[2], Shan Ji[3]*

[1] School of Cyberspace Security, Shandong University of Political Science and Law, China
[2] School of Computer Science, Qufu Normal University, China
[3] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China
leilishang@163.com, wangyijia1027@163.com, dongxinqi1222@163.com, peiyaon@163.com, shanji@nuaa.edu.cn

Abstract

With the unprecedented growth of the tourism industry and the Internet, word-of-mouth about tourist attractions has become an invaluable reference factor. Currently, attractions receive lots of redundant data from visitor reviews, which is time consuming and tedious to read and analyze. This paper attempts to find a correlation between user reviews and the attractiveness of attractions. Therefore, by finding a suitable model to classify users' comments in terms of sentiment and analyzing the results, the scenic spot can better understand the users' evaluation and feedback, so that it can adjust and improve the scenic spot's service in a timely manner. In this paper, we compare the accuracy of sentiment classification between the fastText model and Long-Short Term Memory (LSTM) for the same training set under the same conditions. Simulation results demonstrate that the LSTM is accurate and precise in capturing the scenic comments made by the users. Therefore, we use the LSTM to analyze the data for the emotional tendency and to make suggestions for the improvement of scenic spots.

Keywords: Emotional analysis, Long-Short Term Memory, FastText

1 Introduction

User comments play an indispensable role in the information network as the Internet expands and develops. Today, major online platforms are used by an increasing number of computer users to express their opinions and attitudes towards certain products. The 50th Statistical Report on the Development of China's Internet shows that as of June 2022, the size of China's Internet users was 1.051 billion and the Internet penetration rate reached 74.4%, so China has reached full coverage of the Internet construction and the size of users is steadily increasing [1].

User comments play an unparalleled role. However, there is less research on scene reviews. For example, when arriving in a popular city, most out-of-town visitors don't just look at the popularity or ratings of the attractions they choose to visit, and user reviews play a crucial role in guiding users' choices. Electronic word-of-mouth in tourist attractions is a kind of online sharing by internet users, which contains consumers' opinions and suggestions on products, as well as some of their feelings and subjective emotions [2]. Recognizing the economic value of shared information on the web for the transformation of tourist attractions, natural language processing and neural network algorithms have become important tools. Text Sentiment Classification technology is a sub-field of Natural Language Processing, where text data imported by a computer is used by a machine to extract attitudes, evaluations, and opinions from the text using sentiment analysis models or algorithms [3], and classify the sentiments. According to the method of categorizing affective trends, they can be classified as binary affective classification, ternary affective classification and multiple affective classification [4].

There are two main common approaches to sentiment analysis: Lexicon-based and machine learning-based. The first method does not require manual annotation, most sentiment lexicons suffer from insufficient coverage of sentiment words and missing domain words [4]. The second method requires manual annotation. From the point of view of user comments, the use of text sentiment analysis techniques can directly understand the attitudes and needs of users [5], and thus by analyzing the sentiment tendency of user comments, it is of great significance to the needs of users.

1.1 Related Works

Academics offer some solutions to the problem of analyzing affective tendencies. As early as 2002, B. Pang et al. proposed the distributed machine learning method maximum entropy classification to handle huge review datasets, and they targeted the polarity distribution of movie reviews [6]. In 2015, for the impact of consumer sentiment in forums, a lexicographic method was proposed to classify the N-gram features of text sentiment at a fine-grained level, followed by supervised training using an SVM method, which led to better sentiment classification results [7]. Turney attempts to classify the sentiment tendencies of user comments using unsupervised learning methods [8]. Back in 2013, the task of sentiment analysis was to estimate whether a text was positive or negative by analyzing ratings throughout the text and in the comments

*Corresponding Author: Shan Ji; Email: shanji@nuaa.edu.cn
DOI: https://doi.org/10.70003/160792642025072604006

[9]. A sentiment mining algorithm was proposed, which based on previous research and experimentally demonstrated that the method significantly improves the accuracy of the correlation between patronage and ticket sales at tourist attractions [10]. However, as far as the current analysis of the emotional tendency of user reviews is concerned, most of them are focused on Internet e-commerce, and relatively few of them analyze the emotional tendency of scene reviews.

1.2 Motivations and Contributions

Considering the above problems, we will analyze the user reviews of tourist attractions by comparing two sentiment classification models, and use them as a basis for proposing improvement suggestions to the staff of the attractions, so as to improve the service quality and popularity of the attractions.

The main contributions of this paper are as follows:

(1) We select two classical sentiment classification models and train them on crawling datasets respectively, and set indicators to compare the accuracy of sentiment classification of the two models for specific datasets.

(2) We use visualization technology to analyze and discuss the data after sentiment classification, and use it as a basis to give suggestions for improvement of attractions.

1.3 Roadmap

The rest of the paper is organized as follows. In Section II, the basics about the fastText model and the LSTM model are presented. Then, the implementation principles of these two models are outlined in Section III. In Section IV, we conduct simulation experiments and process and analyze the results. Finally, we summarize this paper in Section V.

2 Preliminaries

2.1 fastText Model

The fastText model uses a hierarchical softmax function to reduce computational complexity [11], and uses the n-gram as the smallest unit. It decomposes each word in the input context into a word-based n-gram format and sums the n-grams of all the decomposed words with the original word to represent the semantic information of the context [12]. The structure is similar to the CBOW model of Mikolov et al. with three layers [13]. The structure is shown in Figure 1. The structure of the fastText model consists of an input layer, a hidden layer and an output layer. The difference between the fastText model and the CBOW model is that the CBOW model uses the contextual feature words introduced in the input layer to predict the surrounding central words, while the fastText model introduces all feature words of a particular document into the input layer [14]. The input layer of the fastText model contains N-gram features; the hidden layer is the summation and averaging of the input data; and the output layer is the corresponding labels of the documents. Due to a large number of categories and the amount of data, the fastText model uses a hierarchical softmax to optimize the speed of the model in the final output.

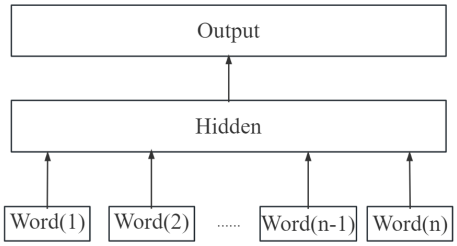


Figure 1. FastText structure

Among them, Word (1), Word (2), Word (n-1), Word (n) are the input words.

2.2 LSTM (Long-Short Term Memory)

Long-Short Term Memory has a strong local perception ability, which can learn text features in natural language processing with ultra-high efficiency and store relevant features in a short period of time to prevent them from being missed when learning new features later [15]. The LSTM model can effectively capture semantic associations between long sequences and its basic structure can be divided into forget gate, input gate, cell state, and output gate. This structure can enable recurrent neural networks to effectively utilize long-distance time series information [16]. The LSTM structure is shown in Figure 2.

Forget Gate: It stitches the input data x_t at time t with the output h_{t-1} from the previous time unit into the neural network layer, and the value of the output f_t is 1 or 0, where 1 means that all the information from the previous unit is retained, and 0 means that the information of the previous unit is all forgotten.

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

Input Gate: The sigmoid function neural network layer gets a vector i_t that determines the information to be updated at the current time. The Hyperbolic Tangent neural network layer generates a vector $\tilde{C}_t$ that is added to the cell state. Multiply i_t and $\tilde{C}_t$, whose value determines how to update the cell state.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_f). \quad (2)$$

$$\tilde{C}_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_C). \quad (3)$$

Cell State: Update cell state with data from the Forget Gate and Input Gate.

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t. \quad (4)$$

Output Gate: The input data x_t at time t, the output h_{t-1} of the previous time unit and the cell state together determine what information to output. The vector O_t of the Output Gate is obtained by passing the sigmoid function

of the old state of h_{t-1} with the data x_t input at moment t through the neural network layer, determining with what probability we should derive the data. The cell state C_t is dotted with the O_t in tanh neural network layer to obtain the result h_t.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o). \quad (5)$$

$$h_t = o_t * tanh(C_t). \quad (6)$$

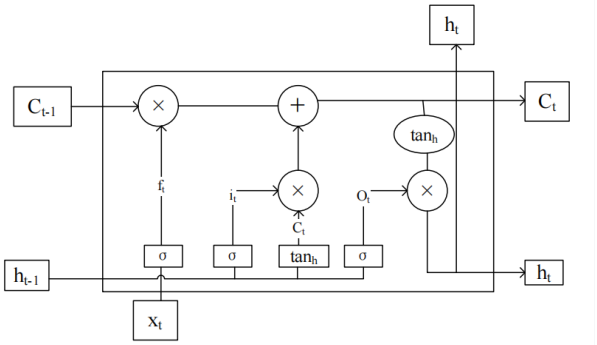


Figure 2. LSTM structure

3 The Principle and Implementation of Text Sentiment Classification Algorithm Based on fastText and LSTM

3.1 FastText-based Text Sentiment Classification

Sentiment classification using the fastText model starts by reading the dataset. The data is then segmented and stored in the segment column. Next, the deactivated words file is read and the words in it are read into a list. Then the data set is traversed and the deactivated words are removed from its split result. Next, the data is split into training and validation sets. The target variable is the score column, representing the sentiment score for each comment. This process resets all indices and labels of both the training set and the data set to start from 0.

The next step is to create two text files containing pre-processed and tokenized data. Each line of these two files will consist of a text sample and a label in the format '_lable_n', where n represents the value of the sentiment score (0 or 1) for each text sample. By calling the join method, the list of tags in each text example is concatenated into a string separated by space characters, which serves as input to the fastText model.

A get_lable function is then defined to extract the predicted labels from the prediction results received from the fastText model, returning a tuple of the form ([labels], []). This works as follows:

Pred is a tuple containing two elements, Pred[0] and Pred[1]. Where Pred[0] is a list of labels for the input text, each prefixed with _label_. For example, if the predicted labels are ['0', '1'], pred[0] will be ['_label_0', '_label_1']. Pred[1] is a list of confidence scores for the corresponding label. If, e.g., predicted confidence scores are [0.8,0.2], then pred[1] is [0.8,0.2]. Then, np.argmax(pred[1]) will find the index of the maximum confidence score in pred[1]. pred[0][index][-1] returns the last character of the predicted label, which is the actual label. If, e.g., pred[0][0][-1] equals _label_1, pred[0][0][-1] equals 1. In the end, the label will be converted to an integer by the int method, and the result will be an integer label.

The predictions returned by the fastText model are then accepted by defining a function that returns the probability that the labels are positive. Pred's input is a tuple of two lists. The first list contains the predicted label, _lable_0 or _lable_1, and the second list contains the probability of the predicted label. The function first creates a dictionary where the keys are the predicted labels and the values are their corresponding probabilities. It then returns the probability that the label is positive (i.e. _lable_1) by indexing the dictionary with the key '_lable_1'.

Finally, the fastText library is used to train a supervised text classification model using the previously generated training data, and the parameters of the training model are initialized. The trained model is then used to classify the sentiment of the test set, and the predicted labels are output along with the corresponding probabilities. Using scikit-learn's accuracy_score function, the predicted labels are compared to the true labels and the performance of the model is evaluated by calculating the accuracy. The main code used to train the model and calculate accuracy is shown in Figure 3.

```
model = fasttext.train_supervised(input="d:/train_semantic.txt",lr=0.1, epoch=100, wordNgrams=3, dim=300)
test_pred = []
for i in range(len(test_data)):
    r = model.predict(" ".join(test_data[i]),k=2)
    test_pred.append(get_label(r))
acc = accuracy_score(test_pred,test_label)
```

Figure 3. Main code for fastText model training and accuracy calculation

3.2 LSTM-based Text Sentiment Classification

To perform text sentiment classification using LSTM, the datafile is first read, split into training, validation and test sets, and converted into a list of tuples.

Next, create a PaddlePaddle data loader object, which takes a dataset object, a conversion function that converts the data samples into operating modes (including training, validation or test), batch sizes and a function that generates small batches of data by merging the sample list. If a conversion function is provided in the create_dataloader function, it will convert the dataset. A batch collector is created based on the operating mode and batch size. Finally, a dataloader object is created using dataset, sampler, batchify_fn.

Then a SelfDefinedDataset class is defined to instantiate three datasets, the training, test and validation sets. Next, the three data loaders are adapted to batch process the training, validation and test sets according to the set batch size. Each sentence in the dataset is padding to the maximum length of the text in the current batch. When defining the data loader object, a data transformation function is set up to convert the read-in data into the data format required by the model, and a batching function forms multiple samples into a batch and padding is

performed after this.

Subsequently, the LSTMModel class is defined to implement the LSTM-based text classification model. The model parameters are first initialized, and a forward method is defined to return the classification probability value given text and length. Then an LSTMModel object is instantiated and passed to paddle.Model, thus creating a model object based on the PaddlePaddle framework.

Next, using the Adam optimizer in the PaddlePaddle framework, pass in all the trainable parameters in the LSTMModel class that the optimizer needs to update and set the learning rate. The cross-entropy loss function is used to calculate the loss value of the model and the accuracy is used as the evaluation metric of the model. Finally, the optimizer, cross-entropy loss function and evaluation metrics are passed in as parameters by calling the model.prepare function to initialize the model setup, train the model and get the accuracy of the model.

To summarize, the paper provides a comparison of the granularity, advantages and disadvantages of the two models above. Table 1 shows the results of the comparison.

Table 1. A summary of the comparison between the two methods

Model	Particle size	Advantages	Disadvantages
fastText	Word level or character level	Based on neural networks, it can learn the contextual information of the text and train faster. It does not require feature extraction and works better. Each word or character is represented as a vector and these are then combined into a text vector for classification.	Using the bag-of-words idea, there is no word order information and limited access to semantic information; the model has the lowest accuracy rate for tourist attraction reviews.
LSTM	Word level or character level	Based on a neural network, it can learn contextual information of the text, handle more complex sentiment analysis tasks and it has a higher accuracy rate for tourist attraction reviews. Each word or character is represented as a vector and then these vectors are combined into sequence vectors for classification.	It has the longest training time and the largest model size.

4 Simulations and Analysis of Results

In this paper, we first use python web crawler to collect the data, and then use jieba participle on the content of the comments, and de-deactivated words on the content and other operations for data preprocessing. Next, the dataset is divided into a training set and a test set, and the training set is trained using two sentiment classification methods, fastText and LSTM, respectively. The model performance is evaluated by the accuracy rate, so that the method with the highest accuracy rate is selected, and the comment data is then sentiment classified and analyzed for the classified data. The specific process is shown in Figure 4.

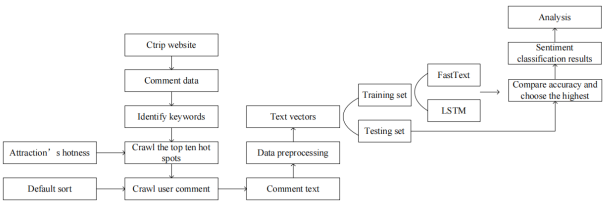


Figure 4. Research flow chart of user review emotional tendency analysis

4.1 Data Collection

Sentiment tendency analysis mainly focuses on the content of user comments, and the experimental dataset selected in this paper comes from the user comment data of Chengdu and Chongqing attractions crawled by python crawler on Ctrip website. The crawled data of the top ten popular attractions in Chengdu and Chongqing contains 10000 items, with positive and negative comments accounting for 73.6% and 26.3% respectively.

4.1.1 Attraction Details

When the web crawler crawls the reviews of each attraction, it also saves the details of the crawled attraction to a local file, and the contents of the main fields of the information are shown in Table 2. Besides, the number of reviews and ratings of the attractions are counted and only the top 30 are listed here, as shown in Figure 5 and Figure 6.

Table 2. Main fields of attraction details

data	pandas dtype
Name of the attraction	object
Evaluation score	float64
Number of reviews	int64
Price	float64
Opening hours	object
Features	object
.....	

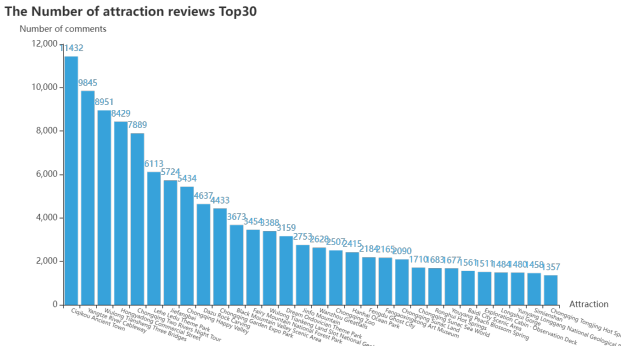


Figure 5. Top 30 spots with the highest number of comments

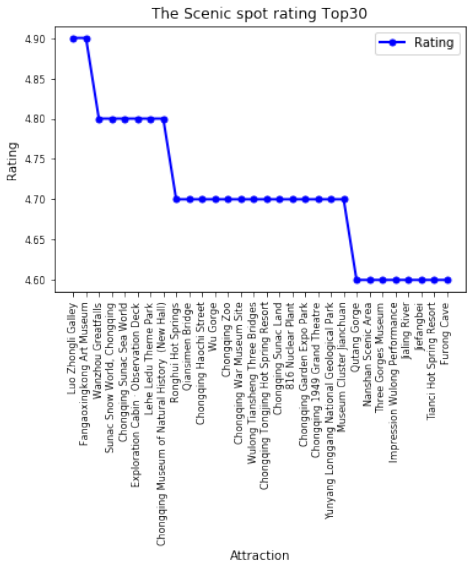


Figure 6. Top 30 highest rated attractions

4.1.2 Content of User Comments

We crawl the reviews of popular scenic spots in Chongqing and Chengdu cities by Python crawler. The review information includes the id of the user, the content of the review, the time of the review and the amount of likes of the review. The specific fields of the comments are shown in Table 3.

Table 3. Main fields of user comments

Id(float64)	comment(object)	Time(object)	Support(int 64)
6511177.0	It is convenient and fast. It is convenient for the county to have buses passing through the scenic spots, but the bus is a little shabby, and the background pattern of 10 yuan is quite spectacular, ...	/Date(1680612563000+0800)/	0
44844665.0	The "White Emperor City" is located on the northern bank of the Yangtze River at the mouth of the Fengjie Qutang Gorge. It was built by Shu Gongsun in the late Western Han Dynasty, because white smoke is often seen in a well...	/Date(1671205702000+0800)/	2
69348055.0	The skiff has passed ten thousand mountains, it is a nice place and worth visiting. The scenery is unique, the cultural landscape is unique, and the water of the Yangtze River goes east.	/Date(1674866722000+0800)/	0

4.2 Data Preprocessing

This paper selects the review data of popular scenic spots in Chongqing and Chengdu as a sample set. In order to make the obtained data of each scenic spot consistent, the obtained scenic spot data must meet the following standards: first, there is a scenic spot review record on the Ctrip website; second, there is a 0-5 point difference in the user evaluation content of the scenic spot; third, the scenic spot is highly popular and has many tourists.

In order to obtain the user review content of each attraction, this paper uses a self-developed python web crawler program to collect the user review content data of the attraction on the Ctrip official website. User reviews of attractions on Ctrip's webpage mainly include four parts: user ID, review content, rating time and number of likes.

Also, the web crawler saves the details of an attraction to the pos.csv file whenever it crawls an attraction. Today's real-world data is very susceptible to noise, missing values, and inconsistent data [17], resulting in poor data quality that will lead to poor computer processing speed and low accuracy of results. It is a critical task to pre-process the data to improve the quality of the data and hence the quality of the test results.

In this paper, firstly, all the records with the same column value in the data records are deleted in the user comments, the words such as English and numbers are removed, and the text is subjected to the operations of word splitting, lexical annotation and removal of deactivated words.

4.3 Comparison and Analysis of Sentiment Classification Models

In this paper we use a machine learning based approach to categorize the sentiment of the collected user review data. Since different models and algorithms are chosen and they perform differently when processing the same dataset, it is crucial to evaluate the chosen model or algorithm. In the binary sentiment classification problem, a sample can be classified into four categories to obtain a confusion matrix as shown in Figure 7.

As shown in Figure 7, True Positives means that the sample is positive and the prediction is positive; False Positives means that the sample is negative and the prediction is positive; False Negatives means that the sample is negative and the prediction is negative; and True Negatives means that the sample is positive and the prediction is negative.

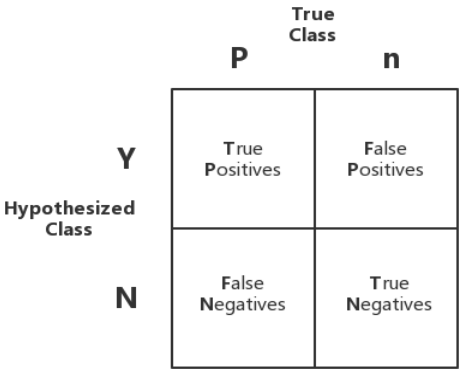


Figure 7. Confusion matrix

This experiment takes the accuracy rate of different models for the same training set as an evaluation index. In this paper, we compare two sentiment classification methods, train the model on the same dataset respectively, test the accuracy rate of the model, select the model with the higher accuracy rate and then classify the data into sentiment classification, divide the obtained data results into positive dataset and negative dataset, and then visualize and analyze the data, which makes the analysis results more accurate.

The confusion matrix shown in Figure 7 is used to further define the scoring metric, accuracy rate. Accuracy rate is a commonly used evaluation metric in sentiment classification. In this paper, the accuracy rate of the test is used to compare different models or algorithms, as in equation (7). Then, the model with a high accuracy rate is used to further analyze the data.

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

In general, the higher the accuracy, the better the model or algorithm is suited to analyze that dataset and the better the results.

In this experiment, the same dataset is trained according to the number of iterations, word vector dimension, and learning rate of fastText model and LSTM, respectively. The data is imported using pandas and numpy libraries, and the word segmentation of the attraction user comment text is performed using jieba segmentation, and the cut text is saved to cut_comment. Then the tone auxiliaries, adverbs, prepositions, conjunctions, etc. are removed using a deactivation word list integrated by multiple parties. The words of the dataset text are converted into a word

frequency matrix and features are extracted by using the CountVectorizer class. In machine learning, the focus of the algorithms used is on the weights of the feature data, hence feature extraction is important [18]. The cut_comment obtained with the help of jieba participle is set to X and the manually labeled score in the training set is set to y. The dataset is divided into a training set and a test set. The two sentiment classification methods are used to train the model using the training set respectively, and then the trained model is selected to analyze the test set to test the accuracy of the model. The comparison results are shown in Table 4.

Table 4. Sentiment analysis accuracy of the two methods

Model / Evaluation Indicator	Learning Rate	Number of Iterations	Word Vector Dimension	Accuracy
fastText	0.1	100	300	0.910
LSTM	5e-5	100	128	0.921

According to Table 4, LSTM has the best accuracy rate of 92.1% for the classification results of user comments of tourist attractions. Therefore, in this paper, we choose LSTM with higher accuracy rate and the best comprehensive effect for the scenic spot user comment test to classify the user comment data of the Ciqikou scenic spot for sentiment classification. First read the review data for the Ciqikou scenic spot and use the trained model to obtain the sentiment score. Assign data with a score greater than or equal to 0.5 to 1, indicating positive, and data below 0.5 to 0, indicating negative, and write the results to a new score column. And the obtained classification results are saved to the local disk.

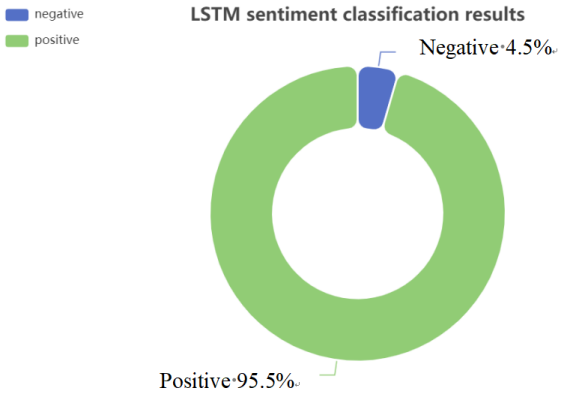


Figure 8. Percentage of LSTM sentiment classification score

After the sentiment classification of the user comments of the Ciqikou scenic spot, this paper saves the positive and negative comments to positive.txt and negative.txt files respectively through the excel table, and carries out the data preprocessing, lexicography, feature extraction for the comment content respectively, and visualizes the user's evaluation of the attraction with the help of word cloud map of WordCloud library in python. The LSTM classification results and word cloud are shown in Figure 8, Figure 9 and Figure 10.

As can be seen from Figure 8, the attraction Ciqikou has a high overall positive review rate. But from Figure 9 and Figure 10 can be seen in the positive comments, the user in the Ciqikou scenic spot for the ancient city, magnets, snacks, twist and other products of high concern, and is very concerned about the characteristics of the scenic area, the overall seems to be that users generally think the attraction is good, worth travelling. In the negative reviews, users believe that the attraction is currently suffering from the commercialization of the old town, the product is more expensive and pitiful, making the user reviews negative. Therefore, in the process of tourism development and construction, scenic spot managers must pay attention to the cultural output of attractions, improve the cost-effectiveness of products sold, nearby hotels should strike a balance between price and comfort, promote the rich development of children's industry, and achieve a win-win situation to satisfy the needs of users and promote economic development.

Figure 9. Positive comments

Figure 10. Negative comments

5 Conclusion

This study compares the performance of two sentiment classification models for sentiment analysis and provides an in-depth analysis. Comparative experiments allow the performance of the two sentiment classification models to be evaluated and help to select the most appropriate model for a given dataset. The fastText model was compared with LSTM using model prediction accuracy as the evaluation

criterion, and it was found that LSTM performed better on the given dataset task. However, there are some limitations. Current sentiment classification models in existing studies only analyze positive and negative sentiment in reviews, fail to identify finer sentiment trends, and lack in-depth analysis of the specific causes and effects of sentiment. Therefore, model fusion methods can be considered in further research to further improve the accuracy of sentiment classification.

Acknowledgment

This research was financially supported by the Scientific Research Project of Shan Dong University of Political Science and Law (2019Z01B), the Teaching Reform Project of Shandong University of Political Science and Law (2018JGA002).

References

[1] China Internet Network Information Center, The 50th Statistical Report on Internet Development in China, *Journal of The National Library of China*, Vol. 31, No. 5, pp. 12-12, October, 2022.

[2] N. Donthu, S. Kumar, N. Pandey, N. Pandey, A. Mishra, Mapping the electronic word-of-mouth (eWOM) research: A systematic review and bibliometric analysis, *Journal of Business Research*, Vol. 135, pp. 758-773, October, 2021.

[3] F. Yuan, S. Chen, K. Liang, L. Xu, *Research on the coordination mechanism of traditional Chinese medicine medical record data standardization and characteristic protection under big data environment*, Shandong People's Publishing House, 2021. ISBN: 978-7-209-13618-1

[4] G. Xu, Y. Meng, X. Qiu, Z. Yu, X. Wu, Sentiment Analysis of Comment Texts Based on BiLSTM, *IEEE Access*, Vol. 7, pp. 51522-51532, April, 2019,

[5] L. Yuan, *Analysis of User Emotional Tendency of YouTube International Chinese Learning Video*, Master's Thesis, Beijing Foreign Studies University, Beijing, China, 2022.

[6] B. Pang, L. Lee, S. Vaithyanathan, Thumbs up? Sentiment classification using machine learning techniques, *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, Philadelphia, PA, USA, 2002, pp. 79-86.

[7] C. Homburg, L. Ehm, M. Artz, Measuring and managing consumer sentiment in an online community environment, *Journal of Marketing Research*, Vol. 52, No. 5, pp. 629-641, October, 2015.

[8] P. D. Turney, Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, 2002, pp. 417-424.

[9] E. Cambria, B. Schuller, B. Liu, H. Wang, C. Havasi, Knowledge-Based Approaches to Concept-Level Sentiment Analysis, *IEEE Intelligent Systems*, Vol. 28, No. 2, pp. 12-14, March-April, 2013.

[10] R. Y. K. Lau, W. Zhang, W. Xu, Parallel aspect-oriented sentiment analysis for sales forecasting with big data, *Production and Operations Management*, Vol. 27, No. 10, pp. 1775-1794, October, 2018.

[11] A. Alessa, M. Faezipour, Z. Alhassan, Text Classification of Flu-Related Tweets Using FastText with Sentiment and Keyword Features, *2018 IEEE International Conference on Healthcare Informatics (ICHI)*, New York, New York, USA, 2018, pp. 366-367.

[12] A. Yin, Y. Wu, Y. Zheng, X. Yu, Base on FastText model to improve the word embedding of phrases and morphology, *Journal of Fuzhou University (Natural Science)*, Vol. 47, No. 3, pp. 314-319, June, 2019.

[13] T. Mikolov, K. Chen, G. Corrado, J. Dean, *Efficient estimation of word representations in vector space*, arXiv, January, 2013. https://arxiv.org/abs/1301.3781

[14] S. Yan, *Research on Text Classification Based on Improved TF-IDF and FastText Algorithm*, Master's Thesis, Anhui University of Science and Technology, Huainan, China, 2020.

[15] M. Fu, L. Pan, Sentiment Analysis of Tourist Scenic Spots Internet Comments Based on LSTM, *Mathematical Problems in Engineering*, Vol. 2022, Article No. 5944954, July, 2022.

[16] Y. Chai, Research on sentiment analysis of book review text based on LSTM and Word2vec, *Information technology*, Vol. 368, No. 7, pp. 59-64+69, July, 2022.

[17] F. Xiao, *Research on sentiment analysis of e-commerce reviews based on latent dirichlet model*, Master's Thesis, Beijing University of Chemical Technology, Beijing, China, 2017.

[18] H. Hung, J. Chen, Y. Ma, Machine Learning Approaches to Malicious PowerShell Scripts Detection and Feature Combination Analysis, *Journal of Internet Technology*, Vol. 25, No. 1, pp. 167-173, January, 2024.

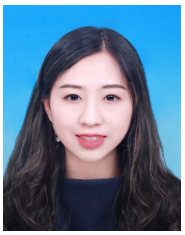
Biographies

Lei Shang, received her master's degree in software theory and engineering from Shandong University. She is currently an associate professor of Shandong University of political science and law. She mainly engages in research on network security and machine learning.

Yijia Wang is currently pursuing her master's degree at the School of Computer Science, Qufu Normal University, Rizhao, Shandong Province, China. She received a bachelor's degree in Data Science and Big Data Technology from Qufu Normal University in 2023. Her research interests include blockchain technology and applications and machine learning.

Xinqi Dong is currently studying for a master's degree at the School of Computer Science, Qufu Normal University, Rizhao City, Shandong Province, China. She received a bachelor's degree in information and computational science from Qufu Normal University in 2022. Her research interests include information security theory and modern cryptography.

Peiyao Niu is currently studying for a master's degree at the School of Computer Science, Qufu Normal University, Rizhao City, Shandong Province, China. He received a bachelor's degree in Network Engineering from Qufu Normal University in 2023. His research interests include blockchain technology and applications and federated learning.



Shan Ji is a master student in Computer Science and Technology, Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, China. Her current research interests include data information security and network security.