WEP Warwick Evans Publishing

International Journal of Computer Science and Information Technology

ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 4, Number 3, Year 2024
DOI: https://doi.org/10.62051/ijcsit.v4n3.17
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Research on Sentiment Analysis Based on Ensemble Learning

Kaiyu Li [a], Yi Xie [b], Shicheng Li [c], Zhaomin Liu [d], Di Liu [e, *]

School of Information Engineering, Wuhan Business School, Wuhan, 430056, China
[a] 2227966941@qq.com, [b] xy3387366267@outlook.com, [c] lzm20050915@outlook.com, [d] 3438970684@qq.com, [e] 63330350@qq.com

ABSTRACT

The purpose of this paper is to use the ensemble learning method to analyze the sentiment of tourist attraction reviews, so as to improve the reference value of tourists in travel decision-making. With the popularity of online reviews, how to extract emotional information efficiently and accurately has become an important topic. In this study, random forest, gradient boosted tree (GBDT) and support vector machine (SVM) were used as the base learners, and the ensemble strategy of weighted voting was used to construct the model. Optimize parameters with cross-validation and grid search, and comprehensively evaluate model performance with metrics such as AUC, Log Loss, MCC, and average accuracy. Experimental results show that the classification accuracy and robustness of the ensemble model are better than those of the single model, with an AUC of 0.92 and an MCC of 0.78. The balance between precision and recall was demonstrated through performance surface plot analysis based on different thresholds, and the optimal threshold range was determined. The model can provide reliable sentiment insights for tourism platform users, help scenic spot managers identify user needs, optimize services, and have scalability in other review analysis fields.

KEYWORDS

Sentiment analysis; Integrated learning; Random forest; Gradient boosting tree; Support vector machines

1. INTRODUCTION

With the development of the internet, the travel industry has become increasingly reliant on user reviews on online platforms. Tourists' evaluation of attractions not only helps others make travel decisions, but also provides valuable advice for scenic spot managers to improve their services. However, in the face of a large number of reviews, how to effectively extract valuable emotional information from them has become a difficult problem. Traditional sentiment analysis methods often rely on dictionaries and rules, but struggle to cope with the complexities of natural language. In order to improve the accuracy of sentiment classification, automated analysis methods based on machine learning and ensemble learning have attracted extensive attention in recent years.

In this study, we propose a sentiment analysis method based on ensemble learning, which constructs a comprehensive sentiment analysis system by combining three base learners: Random Forest, Gradient Boosted Tree (GBDT) and Support Vector Machine (SVM). Compared with a single model, the ensemble learning method can effectively combine the advantages of multiple models, reduce classification bias and variance, and improve the robustness of the model.

Research methods include data collection, data preprocessing, model construction and training, and model performance evaluation. Firstly, 50,000 comments about Hubei scenic spots were obtained from Ctrip through a web crawler, and the comment text was converted into feature vectors available

to the model after data cleaning, word segmentation and TF-IDF feature extraction. During the training process, the model parameters are tuned using cross-validation and grid search to ensure that each base learner performs the best under the optimal configuration. Finally, the ensemble strategy based on weighted voting combined with the prediction results of each base learner is used to form the overall classification output.

The focus of experimental design is to verify whether the ensemble model outperforms the single model in the sentiment classification task. In order to comprehensively evaluate the performance of the model, various indicators such as AUC, Log Loss, MCC, average precision (AP), and cross-entropy were used. The experimental results show that the ensemble model is better than the single model in all indicators, especially in AUC (0.92), MCC (0.78) and average accuracy (0.91). This shows that the ensemble model can classify the sentiment information in comments more accurately and stably.

In addition, the study also performed a threshold sensitivity analysis using the precision-recall surface plot and the F1-score surface plot, and the results showed that the F1-score of the model reached the best balance when the threshold was between 0.5 and 0.6.

Through experimental verification, the efficiency and accuracy of the ensemble model can be reflected. The sentiment analysis system can not only provide valuable sentiment insights for travel platform users to help them make more informed travel decisions, but also provide data support for scenic spot managers to optimize services.

2. RESEARCH METHODOLOGY

2.1. Data Collection and Pre-Processing

Data collection is a critical step in sentiment analysis. This paper uses web crawler technology to crawl from a number of well-known tourist websites to obtain the review information of many famous tourist attractions in Hubei Province. The specific steps are as follows:

The first is the selection of crawler tools, this article uses the Python programming language, combined with Beautiful Soup and Scrapy libraries for data scraping. These tools are able to efficiently parse the structure of a web page and extract the information it needs. Secondly, the selection of the target website, the data of a good website is more specific and real. Special fields are then defined, such as comment time, comment content, and so on. Then, by writing crawler scripts, simulating user behavior, visiting the landing page and extracting the required comment data, about 50,000 comments are finally collected, and the captured data is stored in the MySQL database for subsequent processing and analysis.

Because the raw review data may have noise and redundant information, a series of preprocessing steps are required to improve the effectiveness of subsequent model training. In this paper, the data is preprocessed by means of data cleaning, text word segmentation, stop word removal, and sentiment annotation.

2.2. Establishment of the Model

This paper adopts the method of ensemble learning to construct a sentiment analysis model of tourist attraction reviews. By building multiple machine learning algorithms and combining them. The aim is to make the model more suitable for the study's solving.

Before analyzing the text of a review, the text data needs to be converted into numerical features. In this paper, the TF-IDF method is used as the core step of feature extraction.

2.2.1. TF-IDF

$$TF(t,d) = \frac{The\ number\ of\ times\ the\ word\ t\ appears\ in\ document\ d}{The\ number\ of\ occurrences\ in\ document\ d} \tag{1}$$

Thereinto, TF is the word frequency, which indicates the frequency of a word in the document.

$$IDF(t) = \log \frac{\text{N}}{\text{Number of documents containing the word t +1}} \tag{2}$$

Thereinto, $IDF$is the inverse document frequency, which is the total number of N documents.

$$\text{TF} - \text{IDF}(t) = TF(t,d) \cdot IDF(t) \tag{3}$$

In $TF - IDF$ this study, the comment text can be converted into a high-dimensional feature vector, which is convenient for subsequent machine learning models to process.

2.2.2. Random forest model

By constructing a random forest model, multiple decision trees are built for classification processing, and each decision tree is split by selecting features.

$$C = arg\ max\ P(C = c|X) \tag{4}$$

Where C is its output category; c is the candidate category, which represents all possible category labels;

Sample features are randomly selected to generate multiple decision trees. Suppose there are K trees, and the final output is obtained through a voting mechanism:

$$\hat{y} = arg\ max \sum_{k=1}^{K} I(f_k(x) = c) \tag{5}$$

Thereinto, I is a function of the indicator, $\hat{y}$ is the final category of the prediction, K is the number of decision trees in a random forest, $f_k(x)$ represents the prediction result of the k-th decision tree on the input sample x.

2.2.3. Gradient Boosted Tree (GBDT)

GBDT builds the model in a step-by-step optimization manner, with each new tree to correct the errors of the previous tree.The first is the assumption of the loss function. That is, let the loss function be $L(y, F(x))$.The goal is to minimize this loss:

$$min_F L\big(y, F(x)\big) \tag{6}$$

Thereinto, $F$is the prediction function of the model, which represents the output of the current prediction; y is a real label, which represents the actual category label; x is the input feature vector.

In round m, the model output is:

$$F_m(x) = F_{m-1}(x) + \nu \cdot h_m(x) \tag{7}$$

Thereinto, $F_m(x)$ predicts the value of the model at the time of the m iteration; $F_{m-1}(x)$Predicts the value of the model at the time of the first iteration of $m - 1$; ν is the learning rate, which controls the step size of each iteration update; $h_m(x)$ is the m-th tree, which is used to fit the residuals of the previous round.

Calculates the residuals of the current model:

$$r_i = -\frac{\partial L\left(y_i, F(x_i)\right)}{\partial F(x_i)} \quad \forall i \tag{8}$$

Thereinto, $\partial L(y_i, F(x_i))$is the partial derivative of the loss function to the predicted value, which represents the error gradient of the current model on the sample x_i.

2.2.4. Support Vector Machine (SVM)

In order to maximize the interclass spacing, this paper constructs the optimal hyperplane by SVM for classification, that is, the optimal hyperplane equation is:

$$w \cdot x + b = 0 \tag{9}$$

Thereinto, w is the normal vector of the hyperplane, x is the input feature vector, b is the bias constant.

$$min\frac{1}{2}||w||^2 subject\ to\ y_i(w \cdot x_i + b) \geq 1 \quad \forall i \tag{10}$$

Thereinto, $||w||^2$is the modulus square of the normal vector, which represents the complexity of the hyperplane; y_i is the label of the i-th sample, and its value can be 1 or -1; x_i is the eigenvector of the i-th sample.

For indivisible samples, add a relaxation variable:

$$min\frac{1}{2}||w||^2 + C\sum_{i=1}^{N}\varepsilon_i \tag{11}$$

Thereinto, C is a penalty parameter that controls the tolerance for misclassification.

2.2.5. Model integration

After the training of the above models is completed, the weighted voting method is used to realize the model integration. Weighted voting on the prediction results for each model:

$$\hat{y} = argmax\sum_{k=1}^{K} w_k \cdot I(\hat{y}_k = c) \tag{12}$$

Thereinto, w_k is the weight of the k-th base learner, which indicates the weight of the learner in the vote. $I(\hat{y}_k = c)$ is an indication function, when the k-th model predicts to be category c, the function takes a value of 1, otherwise it is 0.

3. EXPERIMENTS

3.1. Experimental Setup

The experimental data was provided by major well-known travel websites, including review data of several famous scenic spots in Hubei Province, and the data contained 50,000 reviews, covering both positive and negative emotional tendencies. The main steps of the experiment include data set division, experimental environment setting, and parameter setting.

Dataset division: The dataset is divided into training set, validation set, and test set at a ratio of 6:2:2. The training set is used for model training, the validation set is used for parameter tuning and selection of the model, and the test set is used for the final performance evaluation of the model.

Experimental environment setting: The experiment was conducted in an environment equipped with an NVIDIA GTX 1080Ti GPU, and the model was built and trained using the TensorFlow framework. The estimated training time is about 10 hours.

Parameter setting: Random forest, gradient boosting tree (GBDT) and support vector machine (SVM) were selected as the base learners. The main parameters of each base learner are tuned in the

experiment using grid search, and the optimal parameters are selected based on the performance on the validation set.

3.2. Model Training

In random forest training, multiple decision trees are constructed on the training set, and the generalization ability of the model is improved by using the methods of bootstrap sampling and random feature selection.

Gradient boosting tree training constructs multiple decision trees by fitting the negative gradient of the loss function step by step, and each new tree is used to correct the error of the previous round.

Vector machine training is supported, and the kernel method is used to map the input data to a high-dimensional space, and the optimal hyperplane is constructed in this space for classification.

In ensemble model training, the prediction results of each base learner are weighted and voted on during the model fusion phase.

3.3. Parameter Tuning

In a random forest, the number of trees is set to 100 to 500 for searching, the maximum depth of the tree is tuned at different depths, such as 5, 10, or 15, and the maximum number of features is set to sqrt, log2, and auto.

In the gradient boosting tree, the learning rate is set to 0.01, 0.05, 0.1 and other different values for searching, the number of trees is selected from 50 to 300, and the maximum depth of the tree is adjusted from 3 to 10 to balance the complexity of the model and the computational overhead.

In the support vector machine, the penalty parameters are searched in the range of 1 to 100; Types of kernel functions, including linear kernels, RBF kernels, and multinomial kernels; The kernel parameters are adjusted to "scale" or "auto" based on the performance on the validation set.

In the ensemble model, different weights are given to each base learner according to its accuracy on the validation set. The weight ratio was 0.4 for random forest, 0.3 for GBDT, and 0.3 for SVM.

3.4. Performance Evaluation

In order to comprehensively evaluate the performance of the model, ROC-AUC, Log loss, Matthews correlation coefficient (MCC), average precision (AP) and classification cross-entropy were used as evaluation indicators.

The ROC curve is plotted on the horizontal axis with false positive rate and true positive rate on the vertical axis, and AUC represents the area under the ROC curve with values ranging from 0 to 1.

$$AUC = \int_0^1 TPR(FPR)\, dFPR \tag{13}$$

Thereinto, TPR is a true positive rate, which indicates the proportion of positive samples that are correctly predicted to be positive; FPR is a false positive rate, which indicates the proportion of negative samples that are incorrectly predicted to be positive.

The logarithmic loss formula is as follows:

$$Log\ Loss = -\frac{1}{N}\sum_{i=1}^{N}[y_i\, log(p_i) + (1 - y_i)\, log(1 - p_i)] \tag{14}$$

Thereinto, N is the total number of samples, y_i is the real label for the i-th sample, p_i is the predicted probability that the model is in a positive class for the i-th sample.

In order to better consider the comprehensive situation of TP, TN, FP and FN, the Matthews correlation coefficient was calculated for comprehensive measurement.

$$MMC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{15}$$

Thereinto, TP is a true example, i.e., the number of samples that are correctly predicted to be positive classes; TN is a true negative example, i.e., the number of samples that are correctly predicted to be negative; FP is a false positive, i.e., the number of negative samples that are incorrectly predicted to be positive; FN is a false negative example, that is, the number of positive class samples that are incorrectly predicted to be negative.

To measure the overall performance of the model at multiple thresholds, this paper calculates the average accuracy for measurement.

$$AP = \sum_n (Recall_n - Recall_{n-1}) \cdot Precision_n \tag{16}$$

Thereinto, $Recall_n$ is the recall rate at the n-th point, which indicates the proportion of positive samples that are correctly predicted to be positive; $Precision_n$ is the precision of the n-th point, which indicates the proportion of samples that are predicted to be positive that are actually positive; n represents the number of sampling points at different thresholds.

To measure the difference between the predicted probability distribution and the true distribution of the model, cross-entropy is computed for evaluation.

$$Cross - Entropy = -\sum_{i=1}^{N} y_i \log(\hat{y}_i) \tag{17}$$

Thereinto, N is the total number of samples, y_i is the real label for the i-th sample, $\hat{y}_i$ is the predicted probability that the model will belong to that category for the i-th sample.

3.5. Experimental Results and Analysis

According to the well-planned and rigorously executed experimental runs, several experiments were carried out in this paper, and the following rich and valuable results were obtained, as shown in the following table:

Table 1. Model evaluation

model	AUC	Log Less	MCC	AP	Cross-entropy
Random forest	0.87	0.45	0.72	0.85	0.48
GBDT	0.89	0.43	0.75	0.88	0.44
SVM	0.85	0.50	0.69	0.83	0.52
Integration model	0.92	0.39	0.78	0.91	0.41

AUC: The AUC of the ensemble model reached 0.92, which was better than that of all single models, indicating that the ensemble model performed better in distinguishing between positive and negative classes and had higher classification ability.

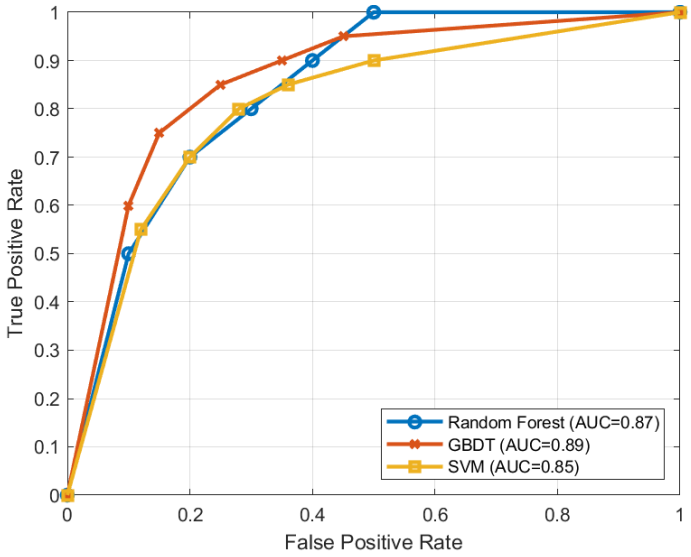

Figure 1. ROC curves of different models

By using professional drawing software (MATLAB) to plot the ROC curve, the AUC value of each model can be clearly observed, and with the help of this method, the classification performance of the model can be seen very intuitively under different thresholds.

Log Loss and MCC: The Log Loss of the ensemble model is 0.39, which is lower than that of other single models. The MCC of the ensemble model is 0.78, which is significantly higher than that of other single models.

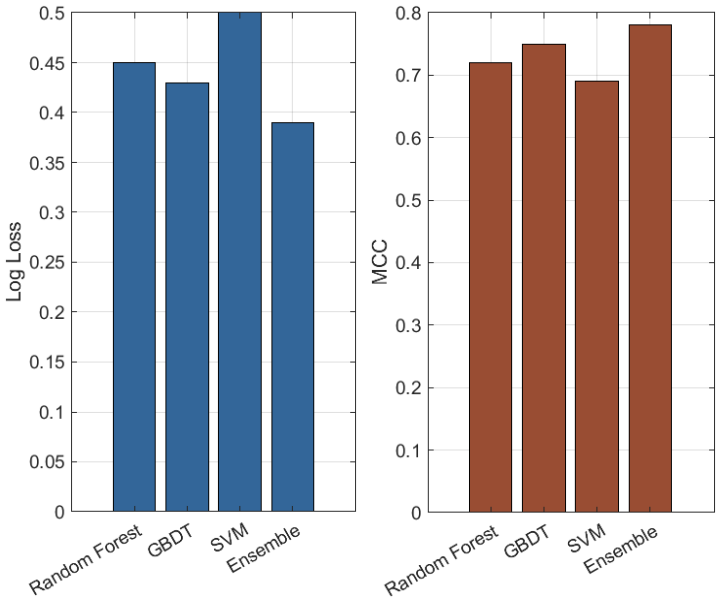

Figure 2. Comparison between Log Loss and MCC

As can be seen in the figure above, the ensemble model has the lowest Log Loss, indicating that it has the best stability and confidence control of the prediction results. In contrast, SVM has a high Log Loss, suggesting that SVM may not be suitable for sentiment classification tasks that can be used directly for probabilistic output. Overall, the comparison results of Log Loss verify the advantages of ensemble learning in improving prediction stability in sentiment classification tasks.

The ensemble model has the highest MCC value, indicating that it has the strongest comprehensive performance in processing sentiment classification tasks. Random forest and GBDT also have high MCC values, while SVM is relatively low, indicating that the ensemble method can effectively

combine the advantages of different models to provide better stability and robustness on unbalanced datasets.

Average Precision (AP) and Cross-Entropy: Through the detailed analysis of Table 1, it is obvious that the ensemble model has a significant advantage in these two coefficients, with an average accuracy of 0.91, which significantly exceeds the accuracy performance of other base learners, reflecting the strong learning and generalization capabilities of the ensemble model. At the same time, the cross-entropy of the ensemble model is only 0.41, which is significantly lower than that of each single model, which further indicates that the ensemble model has excellent data fitting and prediction.

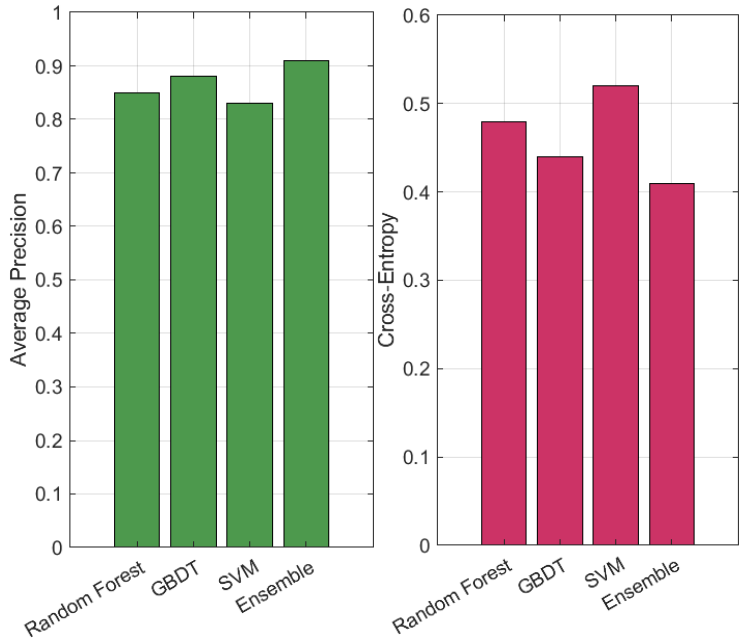

Figure 3. Comparison of average accuracy and cross-entropy

As can be seen in the figure above, the ensemble model combines the advantages of multiple base learners, performs stably in the face of different classification thresholds, and can balance precision and recall well, which is especially suitable for fine sentiment recognition in sentiment analysis tasks. Moreover, the ensemble model can generate more accurate probabilistic predictions in sentiment analysis, which is very important for sentiment recognition in practical applications, because users can make more appropriate judgments based on the confidence level of predictions. The lower cross-entropy also means that the ensemble model is more robust when dealing with complex sentiment data.

In order to further evaluate the performance of the model under different decision thresholds, this paper uses the precision-recall surface plot to show the performance changes of the model. As shown in Figure 4, the PR surface plot shows the correspondence between precision and recall with different combinations of thresholds (0 to 1 range). The graph can help you choose the most suitable threshold for your application to balance classification accuracy and coverage.

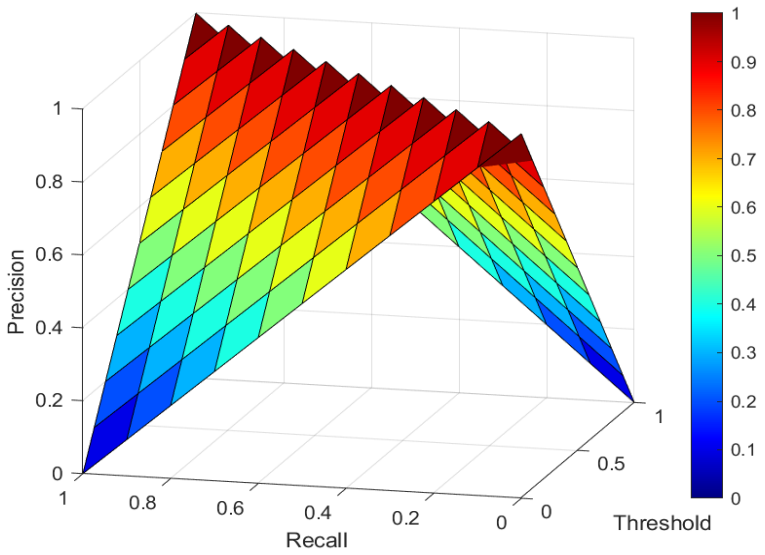

Figure 4. Precision-recall surface plot

As can be seen in the figure above, the higher threshold (near 1) region has a darker color depth, indicating that the accuracy of the model is higher at this time. This means that the model is strict in its predictions of positive class samples, and only when the model is very sure that it is a positive class, will the sample be classified as a positive class. Therefore, although the model predicts positive class samples with high accuracy, this often leads to lower recall, that is, some of the true positive class samples may be misclassified as negative classes. Conversely, at lower thresholds (near 0), the surface color becomes lighter, indicating a decrease in the accuracy of the model. In this case, the model tends to predict more samples as positive, so the recall is higher, but the precision is sacrificed. The performance of this region suggests that the model may increase the coverage of positive class samples, but is prone to introduce more false positives.

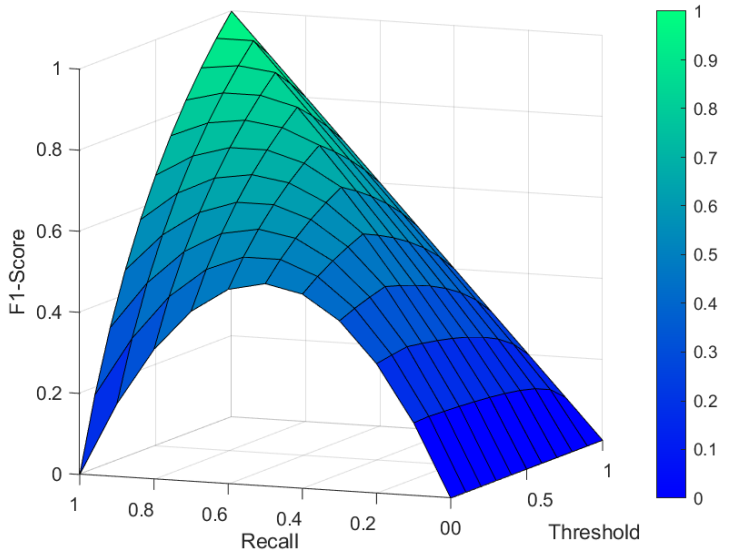

Figure 5. Classification Performance Surface Plots Based on Different Thresholds

As the figure above makes clear, when the threshold is set within the range of 0.5 to 0.6, the model is able to achieve a very good balance between precision and recall. This equilibrium allows the model to exhibit better performance and provide better classification results. It can classify various data more accurately, improve the overall classification accuracy and reliability, and bring greater value and significance to subsequent research and application.

4. CONCLUSION

This paper proposes a sentiment analysis method based on ensemble learning, which combines three kinds of base learners, random forest, gradient boosted tree, and support vector machine, and adopts the ensemble strategy of weighted voting to efficiently classify tourist attraction reviews. In the experimental process, the obtained 50,000 comments were cleaned, segmented and feature extracted, and then the parameters of each base learner were optimized by cross-validation and grid search, and finally the model performance was evaluated by various indicators such as AUC, Log Loss, and MCC. The experimental results show that the AUC of the ensemble model reaches 0.92 and the MCC is 0.78, which is significantly improved compared with the single model. At the same time, the balance between precision and recall and the influence of threshold selection are shown through the threshold-based performance surface diagram. In practical application, the model can provide effective emotional insights for tourism platform users and scenic spot managers, help users make informed travel decisions, and assist managers to quickly locate the pain points in tourist feedback, so as to optimize the service experience.

ACKNOWLEDGEMENT

The authors of this comprehensive document extend their heartfelt gratitude and profound appreciation for the generous funding and invaluable support offered by Project of Collaborative Education between Industry and Education, Ministry of Education: Project Number: 230905181154815. The esteemed 2024 Schol-level Innovation and Entrepreneurship Training Program of Wuhan Business University. (No.202411654174).

REFERENCES

[1] Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval, 2(1-2), 1-135.

[2] Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5-32.

[3] Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of Statistics, 29(5), 1189-1232.

[4] Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273-297.

[5] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830.

[6] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785-794).

[7] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

[8] Aggarwal, C. C., & Zhai, C. (2012). A survey of text clustering algorithms. In Mining text data (pp. 77-128). Springer, Boston, MA.

[9] LIU Bing. (2012). A review of sentiment analysis and opinion mining. Chinese Journal of Computers, 35(6), 1125-1138.

[10] Haiyan Wang, Wei Li, & Liang Zhang. (2018). Textual sentiment analysis of tourist attraction evaluation based on machine learning. Computer Engineering, 44(2), 248-252.

[11] Wu Jiangtao, Wang Wenbin, & Li Cong. (2019). A study on sentiment classification of Chinese reviews based on random forests. Computer Science, 46(2), 84-88.

[12] Zhiming Li, Xu Zhang, & Bing Li. (2020). Application of gradient boosting decision tree in text classification. Journal of Computer Application Research, 37(4), 1023-1028.

[13] ZHOU Xue, ZHAO Feng. (2017). Review of support vector machines in sentiment analysis. Journal of Software, 28(7), 1879-1894.

[14] Chen Weiming, Zhang Xiaoming, Liu Yang. (2015). Research on text classification method based on ensemble learning. Computer Engineering and Applications, 51(4), 117-120.

[15] HUANG Wei, WANG Pengfei. (2019). Research progress on Chinese sentiment analysis based on machine learning. Journal of Information Technology, 38(4), 33-40.

[16] LI Ping, HUANG Xiaojing, LI Ran. (2019). Sentiment analysis of travel reviews based on the combination of deep learning and traditional machine learning. Computer Science, 46(8), 72-77.