

Emotional Analysis of E-commerce Product Review Data

ZiQi Peng¹, Hao Chang¹, CaiYun Yang¹

¹Department of Computer Science, Anhui University of Finance and Economics, Beng'bu City, Anhui Province 233000

Email address: 3631705373@qq.com

Abstract—With the rapid development of e-commerce, online product reviews have become a crucial information source for both merchants and consumers. Traditional rule-based sentiment analysis methods rely on manually annotated dictionaries or corpora, which suffer from low accuracy and high labor costs. To address these issues, this study proposes a sentiment analysis model combining Adaboost (Adaptive Boosting) and Naive Bayes. The Naive Bayes classifier, known for its simplicity and efficiency in text classification, serves as the base weak learner. Adaboost, as an ensemble learning algorithm, optimizes the model by adjusting the weight of misclassified samples and integrating multiple weak learners into a strong classifier, thereby compensating for the shortcomings of Naive Bayes in handling high-dimensional features and limited training data. In the experiment, the proposed model is compared with three classical Naive Bayes variants (Multinomial Naive Bayes, Gaussian Naive Bayes, and Bernoulli Naive Bayes) using a dataset of Ctrip tourism product reviews. The results show that the Adaboost-Naive Bayes hybrid model achieves an accuracy of 90%, which is 8% higher than Multinomial Naive Bayes (the best-performing single Naive Bayes variant), 33% higher than Gaussian Naive Bayes, and 9% higher than Bernoulli Naive Bayes. In terms of precision, recall, and F1-score for positive and negative review classification, the hybrid model also outperforms the three single models. The model realizes a complete workflow including web data crawling, data preprocessing (sentence segmentation, word segmentation, stopword removal, negative word handling, and part-of-speech tagging), model training, and result visualization. It provides intuitive sentiment analysis results for merchants to improve products and for consumers to make purchase decisions, demonstrating practical application value.

Keywords— E-commerce product review; Sentiment analysis; Adaboost algorithm; Naive Bayes; Ensemble learning.

I. INTRODUCTION

In recent years, the prosperity of e-commerce platforms (such as Taobao, JD.com, and Ctrip) has made online shopping the mainstream consumption mode. According to the China Internet Network Information Center (CNNIC), by the end of 2023, the number of online shopping users in China had reached 860 million, and the scale of e-commerce transactions exceeded 45 trillion yuan. A notable feature of e-commerce platforms is that they allow users to publish product reviews after purchase, which contain consumers' subjective evaluations of product quality, service experience, and usage effects. These reviews are valuable for merchants to adjust product strategies and for consumers to avoid "blind consumption" [1]. However, the massive volume of unstructured review text (e.g., millions of reviews for a single hot-selling product) makes manual analysis inefficient and

error-prone, creating an urgent need for automated sentiment analysis technology.

Sentiment analysis, also known as opinion mining, aims to identify and extract subjective information from text and determine its emotional tendency (positive, negative, or neutral) [2]. Early sentiment analysis methods were mainly rule-based: researchers constructed sentiment dictionaries (e.g., HowNet and BosonNLP) and calculated sentiment scores by matching words in the text with dictionary entries. For example, Pan Zhongying designed a Chinese text classifier based on rule-based Naive Bayes, but this method relied heavily on manual dictionary annotation, and its accuracy dropped significantly when facing colloquial or domain-specific reviews [3]. With the development of machine learning, statistical models such as Support Vector Machines (SVM) and Naive Bayes have been widely used in sentiment analysis. Lan Tianfei's study on e-commerce review sentiment analysis showed that Naive Bayes has advantages in computational efficiency and small-sample training, but it assumes that features are independent of each other, which deviates from the actual semantic correlation between words in text, leading to misclassification of complex sentences [2].

In response to the limitations of single models, ensemble learning has become a research hotspot. Adaboost, as a representative boosting algorithm, improves model performance by iteratively training weak learners and adjusting sample weights. Hu Pan et al. used Adaboost to predict the compressive strength of concrete, proving that Adaboost can effectively enhance the generalization ability of weak models [4]. However, few studies have applied the combination of Adaboost and Naive Bayes to e-commerce review sentiment analysis, and existing studies lack sufficient comparative experiments to verify the advantages of hybrid models.

To fill this research gap, this study proposes an Adaboost-Naive Bayes hybrid sentiment analysis model. The main innovations are as follows: (1) Using Naive Bayes as the base weak learner to leverage its efficiency in text feature processing; (2) Introducing Adaboost to optimize the model, focusing on misclassified samples to improve classification accuracy; (3) Conducting comparative experiments with three classical Naive Bayes variants to systematically verify the superiority of the hybrid model; (4) Building a complete system including data crawling, preprocessing, training, and visualization to ensure practical application.

This paper is structured as follows: Section 2 introduces the theoretical basis of the model, including Naive Bayes

variants and the Adaboost algorithm; Section 3 details the system design, including data preprocessing, model construction, and visualization; Section 4 presents the experimental design, results, and analysis; Section 5 summarizes the research and points out future directions.

II. THEORETICAL BASIS

2.1 Naive Bayes Classifier

The Naive Bayes classifier is a probabilistic model based on Bayes' theorem and the assumption of feature independence [13]. Its core idea is to calculate the posterior probability of a sample belonging to a certain category based on prior probability and likelihood, and select the category with the highest posterior probability as the prediction result.

2.1.1 Bayes' Theorem

For a sample with feature vector $X=(x_1, x_2, \dots, x_n)$ and category set $Y=\{c_1, c_2, \dots, c_k\}$, the posterior probability $P(Y = c_k | X)$ is calculated as:

$$P(Y = c_k | X) = \frac{P(X|Y = c_k)P(Y = c_k)}{P(X)}$$

Where:

- $P(Y = c_k)$ is the prior probability of category c_k , calculated as the proportion of samples of category c_k in the training set;
- $P(X|Y = c_k)$ is the likelihood, representing the probability of feature X appearing under category c_k ;
- $P(X)$ is the marginal probability of feature X , which is a constant for all categories and can be ignored in classification.

2.1.2 Common Variants of Naive Bayes

According to the distribution of feature data, Naive Bayes has three common variants, which are applicable to different scenarios:

Multinomial Naive Bayes: Assumes that features follow a multinomial distribution, suitable for discrete feature data such as word frequency in text. It is the most widely used variant in text classification [16];

Gaussian Naive Bayes: Assumes that features follow a Gaussian distribution, suitable for continuous feature data (e.g., image pixel values). It is rarely used in text sentiment analysis due to the discrete nature of text features [17];

Bernoulli Naive Bayes: Treats features as binary variables (0 or 1, indicating the presence or absence of a word), focusing on whether a word appears rather than its frequency. It is suitable for short text classification with sparse features [18].

2.2 Adaboost Algorithm

Adaboost (Adaptive Boosting) is an ensemble learning algorithm proposed by Freund and Schapire in 1997 [20]. Its

core is to "boost" weak learners with low accuracy into strong learners with high accuracy by adjusting sample weights and learner weights.

2.2.1 Core Steps of Adaboost

- Initialization: Assign equal weights to all training samples $w_{1i} = \frac{1}{N}$ (where N is the number of samples);

- Iterative Training of Weak Learners:

- For the m -th iteration, train a weak learner $h_m(x)$ using the weighted training set;
- Calculate the classification error rate of $h_m(x)$: $\delta_m = \sum_{i=1}^N w_{mi} I(h_m(x_i) \neq y_i)$ (where $I(\cdot)$ is the indicator function);
- Calculate the weight of

$$h_m(x) : \alpha_m = \frac{1}{2} \ln \left(\frac{1 - \delta_m}{\delta_m} \right) \text{ (the smaller the}$$

error rate, the larger the weight);

- Update the sample weights:

$$w_{(m+1)i} = \frac{w_{mi} \exp(-\alpha_m y_i h_m(x_i))}{Z_m} \text{ (where}$$

Z_m is the normalization factor to ensure the sum of weights is 1);

- Integration of Weak Learners: The final strong learners:

$$H(x) = \text{sign} \left(\sum_{m=1}^M \alpha_m h_m(x) \right)$$

Where $\text{sign}(\cdot)$ is the sign function, and M is the number of weak learners.

2.2.2 Advantages of Adaboost in Sentiment Analysis

Focus on Misclassified Samples: By increasing the weight of misclassified samples, Adaboost enables subsequent weak learners to focus on difficult-to-classify cases (e.g., reviews with sarcasm or negative words), which is crucial for improving the accuracy of sentiment analysis;

No Need for Parameter Tuning: Adaboost does not require manual adjustment of complex parameters (e.g., regularization coefficients), reducing the difficulty of model optimization;

Good Compatibility: It can be combined with various weak learners. In this study, it is combined with Naive Bayes to make up for the latter's lack of consideration of feature correlation [24].

III. SYSTEM DESIGN

3.1 Overall Workflow

The e-commerce product review sentiment analysis system designed in this study includes four core modules: data collection, data preprocessing, sentiment analysis, and result visualization. The overall workflow is shown in Fig. 1.

Fig. 1. Overall workflow of the sentiment analysis system

Data Collection: Use Python to crawl review data from the Ctrip tourism product platform, including product ID, review content, rating, and review time;

Data Preprocessing: Process the raw review text through sentence segmentation, word segmentation, stopword removal, negative word handling, and part-of-speech tagging to convert it into structured feature data;

Sentiment Analysis: Train the Adaboost-Naive Bayes hybrid model using the preprocessed data, and classify reviews into positive (label 0) or negative (label 1);

Result Visualization: Display the analysis results (e.g., positive/negative review ratio, keyword word cloud) through a web interface to provide decision support for merchants and consumers.

3.2 Data Preprocessing

Raw review text contains noise (e.g., punctuation, meaningless words) and complex semantic structures (e.g., negative words), which need to be preprocessed to improve model performance. The specific steps are as follows:

3.2.1 Sentence Segmentation and Word Segmentation

Sentence Segmentation: Use the `hanlp.utils.rules.split_sentence` method in the HanLP library to split paragraphs into individual sentences based on Chinese punctuation (e.g., "。", "？", "！"). For example, the input "This is a good product. Have you used it?" is split into ["This is a good product.", "Have you used it?"][5];

Word Segmentation: Adopt the ELECTRA pre-trained model (COARSE_ELECTRA_SMALL_ZH) in HanLP for word segmentation. This model uses "replaced token detection" to identify word boundaries, avoiding over-segmentation of compound words. For example, the input "The quality is good" is segmented into ["The", "quality", "is", "good"][6].

3.2.2 Stopword Removal

Stopwords are words that have no significant impact on semantic expression (e.g., pronouns, conjunctions). This study uses the Harbin Institute of Technology stopword list and Baidu open-source stopword library to filter stopwords. For example, the text "I think this product is very good" becomes "product good" after removing stopwords ["I", "think", "this", "is", "very"][7][8].

3.2.3 Negative Word Handling

Negative words (e.g., "not", "no") can reverse the emotional tendency of sentences. This study uses three methods to handle negative words:

Part-of-Speech Tagging and Dependency Parsing: Use SpaCy to identify the dependency relationship between negative words and modified words, and combine them into a new semantic unit (e.g., "not good" → "not_good");

Rule-Based Method: Define rules for common negative structures (e.g., "not + adjective") to directly splice negative words with subsequent phrases;

BiLSTM Model: Use a bidirectional long short-term memory network to capture the contextual impact of negative words, improving the handling of complex negative sentences [9].

3.2.4 Part-of-Speech Tagging

Use the CTB9_POS_ELECTRA_SMALL model in HanLP to tag the part of speech of each word (e.g., noun (NN), verb (VV), adjective (ADJ)). This helps filter non-emotional words (e.g., nouns) and focus on emotional words (e.g., adjectives). For example, the input "I love natural language processing" is tagged as "I/PN, love/VV, natural/NN, language/NN, processing/VV"[12].

3.3 Model Construction

The Adaboost-Naive Bayes hybrid model is constructed in two stages: training the base Naive Bayes model and optimizing it with Adaboost.

3.3.1 Training the Base Naive Bayes Model

Data Splitting: Use the `train_test_split` function in the scikit-learn library to split the preprocessed data into a training set (80%) and a test set (20%);

Feature Extraction: Use `CountVectorizer` to convert text into a word frequency matrix, where each row represents a sample and each column represents the frequency of a word in the sample;

Model Training: Train Multinomial Naive Bayes (selected as the base model after preliminary experiments) using the training set, and save the model and feature extractor using the `joblib` library (only models with an accuracy of over 0.89 are saved to ensure quality).

3.3.2 Adaboost Optimization

Load Base Model: Load the pre-trained Multinomial Naive Bayes model and feature extractor;

Set Adaboost Parameters: Use `AdaBoostClassifier` in scikit-learn, set the base learner to the Naive Bayes model, the number of weak learners (`n_estimators`) to 1000, the learning rate to 0.46, and the algorithm to SAMME.R (supports probability estimation) [24]. The parameter settings are shown in Table 1;

Model Training and Evaluation: Train the Adaboost-Naive Bayes hybrid model using the training set, and evaluate its performance on the test set using indicators such as accuracy, precision, recall, and F1-score.

TABLE I. Adaboost parameter settings.

| Parameter | Description | Set Value |
|----------------|---------------------------------|---|
| base_estimator | Base weak learner | Pre-trained Multinomial Naive Bayes model |
| n_estimators | Maximum number of weak learners | 1000 |
| learning_rate | Learning rate | 0.46 |
| algorithm | Boosting algorithm | SAMME.R |
| random_state | Random seed | None |

3.4 Result Visualization

Use the Django framework and ECharts library to build a web-based visualization interface, which includes three core functions:

Product Query: Users input the product ID and select the platform (e.g., Ctrip) to query the product's review data;

Emotional Tendency Display: Use pie charts to show the proportion of positive and negative reviews, and bar charts to show the monthly change trend of review sentiment;

Keyword Word Cloud: Generate a word cloud of positive and negative review keywords (e.g., "good quality" in positive

reviews, "poor service" in negative reviews) to intuitively reflect the focus of consumers.

IV. EXPERIMENTAL DESIGN AND RESULT ANALYSIS

Use either SI (MKS) or CGS as primary units. (SI units are strongly encouraged.) English units may be used as secondary units (in parentheses). This applies to papers in data storage. For example, write "15 Gb/cm² (100 Gb/in²).” An exception is when English units are used as identifiers in trade, such as "3½-in disk drive.” Avoid combining SI and CGS units, such as current in amperes and magnetic field in oersteds. This often leads to confusion because equations do not balance dimensionally. If you must use mixed units, clearly state the units for each quantity in an equation.

The SI unit for magnetic field strength *H* is A/m. However, if you wish to use units of T, either refer to magnetic flux density *B* or magnetic field strength symbolized as $\mu_0 H$. Use the center dot to separate compound units, e.g., "A·m².”

V. OTHER RECOMMENDATIONS

4.1 Experimental Data

The experimental data is crawled from the Ctrip tourism product platform using Python's requests and lxml libraries. The crawling process includes:

Simulate Browser Requests: Add the User-Agent and Referer fields in the request header to avoid being blocked by the platform;

Page Turning Processing: Use POST requests to obtain multi-page review data (up to 100 pages per product);

Data Labeling: Use the SnowNLP library to initially label the reviews (score 0 for positive, 1 for negative) and manually correct 10% of the samples to ensure labeling accuracy.

Finally, 5000 valid reviews are obtained, including 3200 positive reviews (64%) and 1800 negative reviews (36%). The data is split into a training set (4000 samples) and a test set (1000 samples).

4.2 Evaluation Indicators

To comprehensively evaluate the model performance, five indicators are used:

- Accuracy (Acc): The proportion of correctly classified samples to the total samples, reflecting the overall classification ability;
- Precision (P): The proportion of truly positive samples to the samples predicted as positive, reflecting the accuracy of positive predictions;
- Recall (R): The proportion of truly positive samples to the samples actually positive, reflecting the ability to identify positive samples;
- F1-Score: The weighted harmonic average of precision and recall, balancing the two indicators to avoid one-sided evaluation;
- Macro Average: The average of indicators across all categories, suitable for imbalanced datasets (such as the positive and negative review ratio in this study).

The formulas for the indicators are as follows:

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

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1-Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Where:

- *TP* (True Positive): Number of samples correctly predicted as positive;
- *TN* (True Negative): Number of samples correctly predicted as negative;
- *FP* (False Positive): Number of samples incorrectly predicted as positive;
- *FN* (False Negative): Number of samples incorrectly predicted as negative.

4.3 Comparative Experiments

To verify the superiority of the Adaboost-Naive Bayes hybrid model, comparative experiments are conducted with three classical Naive Bayes variants (Multinomial Naive Bayes, Gaussian Naive Bayes, Bernoulli Naive Bayes) under the same experimental conditions (same dataset, feature extraction method, and evaluation indicators).

4.3.1 Experimental Results

The performance of each model on the test set is shown in Table 2, and the visualization of key indicators (accuracy, precision, recall, F1-score) is shown in Fig. 2, Fig. 3, and Fig. 4.

TABLE II. Performance comparison of different models on the test.

| Model | Gaussian Naive Bayes | Bernoulli Naive Bayes | Multinomial Naive Bayes | Adaboost-Naive Bayes |
|--------------------|----------------------|-----------------------|-------------------------|----------------------|
| Accuracy | 0.57 | 0.82 | 0.90 | 0.98 |
| Positive Precision | 0.61 | 0.80 | 0.92 | 0.97 |
| Positive Recall | 0.76 | 0.91 | 0.96 | 0.99 |
| Positive F1-Score | 0.68 | 0.85 | 0.94 | 0.98 |
| Negative Precision | 0.46 | 0.85 | 0.88 | 0.99 |
| Negative Recall | 0.29 | 0.68 | 0.82 | 0.96 |
| Negative F1-Score | 0.36 | 0.75 | 0.85 | 0.97 |
| Macro F1 | 0.52 | 0.80 | 0.89 | 0.97 |

Fig. 2. Comparison of accuracy among different models (Note: The horizontal axis represents the model, and the vertical axis represents the accuracy. The accuracy of the Adaboost-Naive Bayes model is significantly higher than that of the other three models.)

Fig. 3. Comparison of F1-score for positive reviews among different models

(Note: The horizontal axis represents the model, and the vertical axis represents the F1-score for positive reviews. The

Adaboost-Naive Bayes model achieves the highest F1-score of 0.98.)

Fig. 4. Comparison of F1-score for negative reviews among different models

(Note: The horizontal axis represents the model, and the vertical axis represents the F1-score for negative reviews. The Adaboost-Naive Bayes model has a F1-score of 0.97, which is much higher than the Gaussian Naive Bayes model (0.36) and Bernoulli Naive Bayes model (0.75).)

4.3.2 Result Analysis

Performance of Single Naive Bayes Variants:

Gaussian Naive Bayes: The lowest accuracy (0.57) and F1-score (0.52) are mainly due to its assumption that features follow a Gaussian distribution, which is inconsistent with the discrete nature of text word frequency features. It is not suitable for text sentiment analysis;

Bernoulli Naive Bayes: The accuracy (0.82) and macro F1 (0.80) are better than Gaussian Naive Bayes, but it only considers the presence or absence of words, ignoring word frequency information. For example, it cannot distinguish between "very good" and "good", leading to lower accuracy for negative reviews (F1-score 0.75);

Multinomial Naive Bayes: The best performance among single models, with an accuracy of 0.90 and a macro F1 of 0.89. It fully utilizes word frequency information, which is more in line with the characteristics of text data. However, due to the assumption of feature independence, it still misclassifies sentences with semantic correlations (e.g., "The product is not bad" is mistakenly classified as negative).

Advantages of the Adaboost-Naive Bayes Hybrid Model:

Higher Overall Accuracy: The accuracy reaches 0.98, which is 8% higher than Multinomial Naive Bayes, 33% higher than Gaussian Naive Bayes, and 9% higher than Bernoulli Naive Bayes. This is because Adaboost optimizes the misclassified samples: in the iterative process, it increases the weight of samples misclassified by the base Naive Bayes model (e.g., reviews with negative words), making subsequent weak learners focus on these difficult samples;

Better Balance of Positive and Negative Classification: For positive reviews, the F1-score reaches 0.98 (1% higher than Multinomial Naive Bayes); for negative reviews, the F1-score reaches 0.97 (12% higher than Multinomial Naive Bayes). This solves the problem that single Naive Bayes models tend to be biased towards the majority category (positive reviews) in imbalanced datasets;

Stronger Robustness: The model can handle complex semantic expressions such as sarcasm and negative words. For example, the review "The product looks good, but it broke after one use" is correctly classified as negative, while Multinomial Naive Bayes mistakenly classifies it as positive due to the presence of the positive word "good".

VI. CONCLUSION AND FUTURE WORK

5.1 Research Conclusion

This study addresses the limitations of traditional sentiment analysis methods (low accuracy, high labor costs) and proposes an Adaboost-Naive Bayes hybrid model for e-commerce product review sentiment analysis. The main

conclusions are as follows:

Model Innovation: By combining Adaboost (ensemble learning) and Naive Bayes (statistical learning), the model makes up for the shortcomings of single Naive Bayes models in handling high-dimensional features and semantic correlations. Adaboost optimizes the model by adjusting sample weights and integrating multiple weak learners, significantly improving classification accuracy;

Experimental Verification: Comparative experiments with three Naive Bayes variants show that the Adaboost-Naive Bayes model achieves an accuracy of 0.98, which is significantly higher than Gaussian Naive Bayes (0.57), Bernoulli Naive Bayes (0.82), and Multinomial Naive Bayes (0.90). It also performs well in terms of precision, recall, and F1-score, especially in classifying negative reviews (F1-score 0.97);

Practical Application Value: The study builds a complete system including data crawling, preprocessing, model training, and result visualization. Merchants can use the system to quickly understand consumer needs and improve products (e.g., optimizing product quality based on negative review keywords such as "easy to break"); consumers can use the system to make informed purchase decisions, reducing the risk of "blind consumption".

5.2 Limitations and Future Work

Although the study achieves certain results, there are still limitations that need to be addressed in future research:

Handling of Sarcastic Reviews: The current model still has low accuracy in classifying sarcastic reviews (e.g., "Great, the product didn't work at all"). Future work will introduce pre-trained language models (e.g., BERT, ELECTRA) to capture contextual semantic information and improve the recognition ability of sarcastic expressions;

Expansion of Dataset Scope: The experimental data is limited to Ctrip tourism products. In the future, we will crawl review data from multiple e-commerce platforms (e.g., Taobao, JD.com) and multiple categories (e.g., electronics, clothing) to enhance the generalization ability of the model;

Optimization of System Response Speed: The current system has a long response time when requesting back-end data (about 3-5 seconds). Future work will optimize the data storage structure (e.g., using Redis for caching) and model inference speed (e.g., model quantization) to reduce the response time to within 1 second;

Introduction of Neutral Reviews: The current model only classifies reviews into positive and negative, ignoring neutral reviews (e.g., "The product is average"). Future work will add a neutral category to make the sentiment analysis results more detailed and accurate.

ACKNOWLEDGMENT

This article was supported by the National Undergraduate Innovation and Entrepreneurship Training Program (Grant No.: 202410378054).

Finally, I thank the reviewers for their constructive comments, which greatly improved the paper's quality and academic rigor.

REFERENCES

- [1]. Yang G, Xiao X L. Design And Implementation Of Online Shopping Website Based On Asp.Net Mvc[J]. Education And Teaching Forum, 2017, (26): 257-258. (In Chinese)
- [2]. Lan T F. Research On Sentiment Analysis Of E-Commerce Product Reviews[D]. Huazhong University Of Science And Technology, 2023. (In Chinese)
- [3]. Pan Z Y. Design And Implementation Of Naive Bayes Chinese Text Classifier[J]. Computer Programming Skills & Maintenance, 2021, (02): 37-39. (In Chinese)
- [4]. Hu P, Xiao Y, Wang F, Et Al. Prediction Of Concrete Compressive Strength Based On Adaboost Algorithm[J]. Journal Of Wuhan Institute Of Technology, 2024, 46(01): 111-118. (In Chinese)
- [5]. An R X. Research On Public Interest Judicial Intelligence Based On Text Mining[D]. Capital University Of Economics And Business, 2022. (In Chinese)
- [6]. Drugai. Iclr 2020 | Electra: A New Text Pre-Training Model[Z]. <https://blog.csdn.net/U012325865/article/details/106485462>, 2020-06-01. (In Chinese)
- [7]. Gao S, Li S J, Cai Z P. A Review Of Chinese Text Classification Based On Deep Learning[J]. Computer Engineering & Science, 2024, 46(04): 684-692. (In Chinese)
- [8]. Zhong Y B, Nong J, Du Y H. Text Classification Analysis Of Agricultural Product E-Commerce Reviews Based On Lda Topic Model[J]. Gansu Agriculture, 2023, (12): 64-68. (In Chinese)
- [9]. Li B J, Xiang X C R, Deji K Z, Et Al. Research On Tibetan Dependency Parsing Based On Bi-Lstm[J]. Computer Simulation, 2023, 40(07): 300-304. (In Chinese)
- [10]. Liu W, Chen L. A New Algorithm For Cpg Island Location Recognition[J]. Journal Of Chinese Computer Systems, 2012, 33(07): 1557-1563. (In Chinese)
- [11]. Tang F. Research On Chinese Negative Information Recognition Based On Deep Learning[D]. Chongqing University, 2018. (In Chinese)
- [12]. Chen X Y, Ni L P, Ni Z W. Research On Financial Event Extraction Method Based On Electra Model And Part-Of-Speech Features[J]. Data Analysis And Knowledge Discovery, 2021, 5(07): 36-47. (In Chinese)
- [13]. Zcz Xiaozhi. Principle Of Naive Bayes Algorithm In Machine Learning + Python Implementation[Z]. <https://blog.csdn.net/Zcz0101/article/details/109577494>, 2020-11-09. (In Chinese)
- [14]. Logistics It Circle. Classic Model Of Data Analysis - Bayes Theory, Explained In 10 Minutes[Z]. <https://cloud.tencent.com/developer/article/1540339>, 2019-11-18. (In Chinese)
- [15]. Wang P, Pan B G, Xu D J. Probability Theory And Mathematical Statistics[M]. Hefei: Hefei University Of Technology Press, 2020. (In Chinese)
- [16]. Xia P. Research On Programming Code Construction Method Based On Multinomial Naive Bayes Algorithm[J]. Informatization Research, 2023, 49(06): 42-46. (In Chinese)
- [17]. Yuan M G, Xin Y, Wang Z C, Et Al. Damage Identification Of Large Steel Truss Girder Bridges Based On Gaussian Bayesian Network[J]. Journal Of Railway Science And Engineering, 2022, 19(11): 3293-3302. (In Chinese)
- [18]. Zhang L J, Xia Y, Cheng X P, Et Al. Research On Prediction Of Poor College Students Based On Bernoulli Naive Bayes Model[J]. Information Technology And Informatization, 2021, (11): 159-161. (In Chinese)
- [19]. Zhou X L, Dai X P, Chen H L. Sentiment Analysis Of Review Text Based On Naive Bayes Model[J]. Science And Technology Innovation, 2021, (33): 88-90. (In Chinese)
- [20]. Kang Y, Ren J, Quan K, Et Al. Research On Terrace Extraction Method Based On Improved Adaboost[J]. Computer Applications And Software, 2023, 40(10): 212-221. (In Chinese)
- [21]. Wu S Q, Nagahashi H. Parameterized Adaboost: Introducing A Parameter To Speed Up The Training Of Real Adaboost[J]. Ieee Signal Processing Letters, 2014, 21(6): 739-743.
- [22]. Rubin D B. A Calibrated Multiclass Extension Of Adaboost[J]. Journal Of Machine Learning Research, 2011, 10(1): 2387-2415.
- [23]. Li X C, Wang L, Sung E. Adaboost With Svm-Based Component Classifiers[J]. Pattern Recognition, 2008, 41(5): 1600-1611.
- [24]. Zhou L A. E-Commerce Sentiment Data Analysis Method Based On Adaboost-Ga Model[J]. Information Technology, 2023, (05): 72-77+83. (In Chinese)
- [25]. Zou H, Li X L, Jin J. Human Detection Algorithm Based On Adaboost Combined With Improved Gaussian Mixture Model[J]. Foreign Electronic Measurement Technology, 2023, 42(09): 19-27. (In Chinese)