LAUTECH Journal of Engineering and Technology 18 (1) 2024: 47-56

APPLICATION OF MACHINE LEARNING TO TEXT CLASSIFICATION

[1*]Ozoh P., [2]Rasheed S., [3]Akanbi C., [4]Olayiwola M., [5]Ibrahim M., [6]Kolawole M., [7]Olubusayo O., [8]Adigun A.

[1,2,3,5,8]Department of ICT, Osun State University, Nigeria
[4,6]Department of Mathematical Sciences, Osun State University, Nigeria,
[7]Department of Physics, Osun State University, Nigeria

Corresponding Author, email: patrick.ozoh@uniosun.edu.ng: olayiwola.oyedunsi@uniosun.edu.ng

ABSTRACT

The information superhighway provides important principles for giving out information to various consultations. Organizations depend on knowing customer observations about products and services. Data can be enormous to process physically. This study investigates a technique applying Python programming to collect datasets instinctively. The use of machine learning models evolves by applying Random Forest and Naïve Bayes algorithms. These techniques are applied to the data collected for text classification purposes. This process distributes data into; positive, negative, slightly negative, slightly positive, or neutral. The results from the study show the Random Forest classifier is more efficient than the Naïve Bayes algorithm, resulting in an accuracy rate of 76.5% about Naïve Bayes (70.01%). This technique enables organizations to receive insights into customer ways of thinking.

Keywords: *Text classification, Internet community, Random Forest (RF), Insight, Data scraping.*

INTRODUCTION

The application of machine learning models to analyze articles was discussed by Rejeb et al. (2024). The study shows that the ChatGPT is an important tool for students and educators. The study indicates ChatGPT's crucial function in improving students' writing duties and enhancing an interactive learning community. The study finds theoretical and practical concerns for applying ChatGPT in educational institutions. Choe et al. (2024) investigate conducting measurable learning by introducing SAMA, which integrates classification algorithms and models. When the SAMA algorithm is compared with large-scale learning benchmarks, SAMA produces a reduction in storage capacity. Also, SAMA-based data optimization produces harmonious enhancements in text classification accuracy. Abubakr et al. (2024) present a relative analysis between two models for a multi-class classification. The result from the study indicates the proposed application of the deep learning technique yielded an accuracy of 94.95%, compared to 85.71% for the previous technique.

Mupaikwa (2024) proposed in digital libraries. The technique utilized the KNearest neighbor, Bayesian networks, fuzzy logic, support vector machines, clustering, and classification algorithms. The paper proposed the training of librarians, curriculum reviews, and research on Python-dependent technology for libraries. Büyükkeçeci & Okur (2024) discuss the feature selection technique for selecting features relevant to machine learning functions. This study focused on feature selection and feature selection stability. This technique minimizes dataset size. This plays a role in improving the performance of machine learning models. Valtonen et al. (2024) proposed a standard research database of unstructured text and encountered the representativeness difference

between collections of preprocessing and UML-based algorithms that confront research undertakings and transparency. The study requires for contextual representations to focus on issues and offer recommendations for addressing contextual suitability of the UML in research settings. A review of past research works on text mining was done by Shamshiri et al. (2024). The paper investigates the aim of conducting several research works having special functions. The findings from this paper will enumerate important insights, resulting in further progress in computer network research and its connection to academia and industry.

Duan et al. (2024) proposed measuring a data set, including social media to integrate with the system's decision-making process. The system process will depend on several types of data collected from elsewhere. The research uses text-mining techniques to process Twitter data. This paper applies Naïve Bayes, Random Forest, and XGBoost techniques to classify comments on social media. The paper uses the sampling method to compute imbalances in class distribution and obtains public opinion about street cleanliness. This research can be applied to other social media platforms, including Facebook. The study can derive costs and get an understanding of the efficiency of the study. Umer et al. (2023) propose the CNN model together with text classification. The technique was applied to the classification model to produce a word-embedded model. In addition, the proposed technique has been applied on Twitter. The system shows the reliability of the Fast Text word embedded system.

A practical framework is presented in Pal et al. (2023). The paper finds solutions to challenges in research by investigating user comments for some websites. The study selects the principal variables known as predictors and classifies the predictors into two groups depending on their relative importance. The results from the study indicate that time, cost, responsiveness, and accessibility are predictors for producing significant user experience on the internet. The recommendations from this research will improve the quality resulting in more user contentedness. Kariri et al. (2023) examine the total study of ANNs and provide directions for future research. The research enumerates several articles and various journals using a text-mining technique. The study indicates that research in machine learning is increasing. The study proposed by Kariri et al. (2023) requires the availability of a framework to provide a robust study for ANNs. Abdusalomovna (2023) presents a framework for the application to examine unstructured text in databases to transform the data into structured data usable for artificial intelligence (AI) technology.

METHODOLOGY

This research utilizes web scraping as a method for data collection, employing a scraper developed in the Python programming language. This approach is chosen over the conventional method of copy and paste due to its efficiency and time-saving capabilities. Web scraping automates the process, enabling the collection of large volumes of data from websites in a matter of minutes, a task that would otherwise be tedious and time-consuming. It's important to note that web scraping is limited to textual comments and does not include animations or images. The research focuses on gathering data spanning five years of reviews on both perishable and non-perishable food products from Amazon's webpage. A total of 113,683 datasets were collected using this method. Random Forest and Naïve Bayes classifiers were selected for analysis, as they are known to perform well with large datasets.

The system architecture is illustrated in Figure 1 which provides an overview of the research framework. The proposed architecture is collected with the help of the web scrapping method, which is later pre-processed (text transformation).

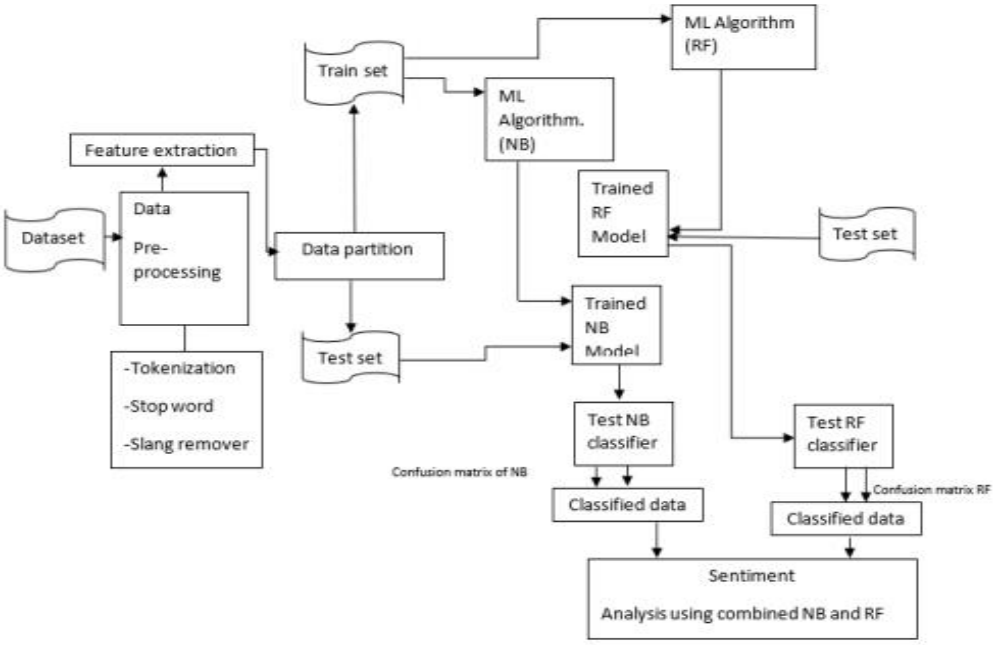

Figure 1. System architecture.

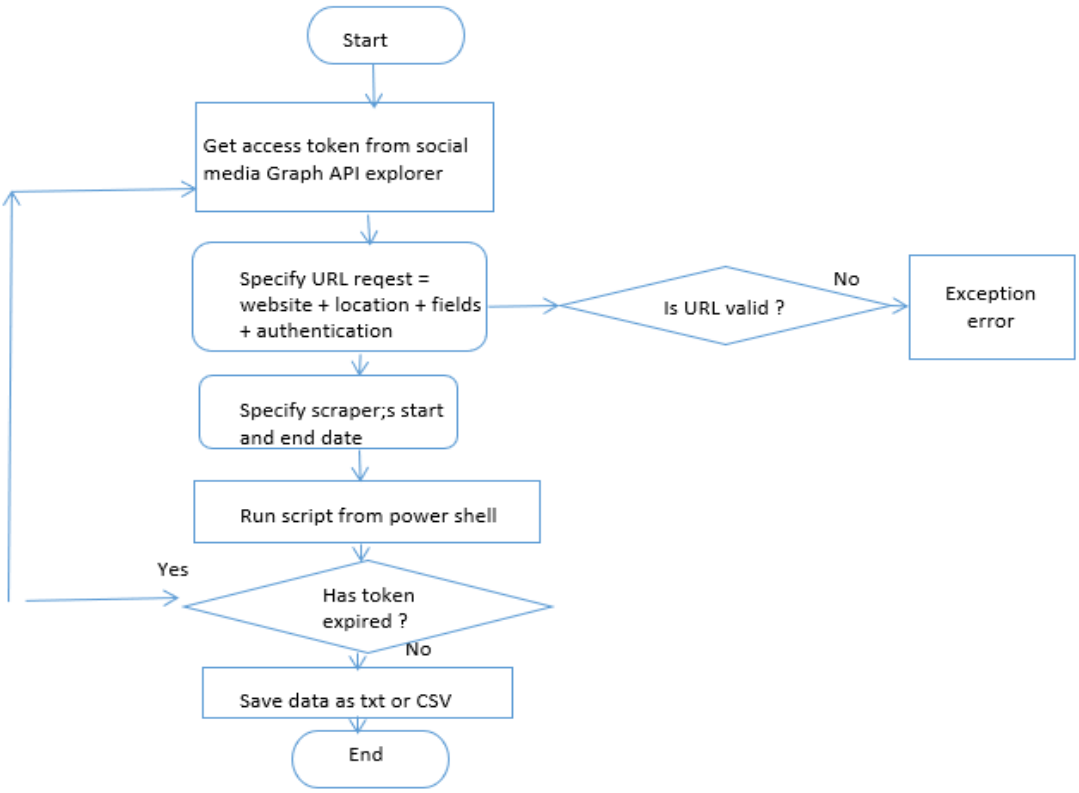

Figure 2. Flowchart depicting web process scrapping.

The dataset is divided into train and test datasets. The train set is input into the algorithm to develop the models. The test set is input into the trained model to predict the results. The output from the

models is analyzed to investigate their performances. Additionally, the flowchart outlining the data collection and analysis process is presented in Figure 2, offering a visual representation of the methodology employed in the study.

Random forest and Naïve Bayes classifiers

The Random Forest classifier selects its output category based on a majority vote, whereby the most frequently occurring category among the predictions from multiple trees is considered as the final result. This approach ensures robustness and reliability in classification. Moreover, Random Forest classifiers are user-friendly, requiring minimal expertise and programming skills. They are accessible to both experts and novices, making them suitable for individuals without an extensive mathematical background.

The Naïve Bayes classifier is a method based on Bayes' theorem. It operates under the assumption that the presence of a particular feature during classification is independent of the presence of other features. This model is particularly advantageous for handling very large datasets due to its simplicity and ease of implementation. In addition to its simplicity, the Naïve Bayes classifier is well-suited for problems that involve associating objects with discrete categories. It belongs to the group of numerically-based approaches and offers several benefits, including simplicity, speed, and high accuracy. Overall, Naïve Bayes classifiers provide a straightforward and efficient solution for a wide range of classification tasks. Spiteri et al. (2020) describes the Bayes rule as:

$$\gamma(\beta) = (\gamma(\alpha|\beta))/(\gamma(\alpha)^{*}\gamma(\beta|\alpha) \quad (1)$$

Where α is the specific class, β is the intended document to be classified, γ(α) and γ(β) are the prior probabilities, γ(α | β) and γ(β | α) are the posterior probabilities. The value of class α might be positive, slightly negative, negative, or neutral.

A review of food products can be considered as a document. Verzi & Auger (2021) highlighted that the multinomial model of Naive Bayes effectively captures word frequency information within documents. The Maximum Likelihood Estimate (MLE) determines the most likely value for each parameter given the training data, thereby providing a reliable ratio. This approach helps in accurately estimating the parameters based on the available training data. For the previous likelihood, this estimate is given as:

$$\gamma(\alpha)=(Nc)/N \quad (2)$$

Where Nc is the total number of documents in class α while N is the total amount of documents. The multinomial model assumes every other given value for the actual class independent of attributes value:

$$\gamma(\beta\}\alpha) = \gamma(\varphi_1 \ldots \varphi_{nd})|\alpha) \quad (3)$$

In the multinomial model, a document is structured as a sequence of word occurrences drawn from the same vocabulary, denoted as V. Each document, denoted as βi, is considered independent of others. The parameter βi represents the distribution of words within each document, following a multinomial distribution with numerous independent trials. This results in the common bag-of-words (BOW) representation for documents. The BOW model is commonly utilized in document classification tasks, where the frequency of word occurrences serves as features for training classifiers. A unigram feature is employed to indicate the presence of a single word within a text interval. This approach enables the representation of documents based on the occurrence of individual words, facilitating effective classification processes.

The conditional probability γ(ω | α) is estimated as the relative frequency in term of ω in documents belonging to class α including multiple occurrences of a term during a document.

$$\gamma(\varphi|\alpha) = (\text{count}(\varphi, \alpha) + 1)/(\text{count}(\alpha)+|V|) \quad (4)$$

Where count (ω, α): Number of occurrences of ω in training documents from class α. count(α): Number of words therein class.

|V|: Number of terms within the vocabulary in the test set

To address the issue of zero probability, the add-one or Laplace smoothing technique is applied, which involves adding one to every count. This adjustment ensures that no probability values are zero. Subsequently, the likelihood of a document given its category is calculated using the multinomial distribution, as presented in Equation (4). Finally, utilizing posterior probability, the new document is classified.

Let αNB represent the posterior probability, where αj is from class α and βi is the ith document. By calculating the posterior probability based on the likelihood of the document given its category, classification of the new document can be achieved effectively.

$$\alpha NB = \arg\max \alpha\, j \in \alpha\pi i(\gamma(\beta i|\alpha j)) \quad (5)$$

Consider Table 1 as the dataset comprising product reviews. The objective of the model is to classify these reviews into either positive or negative categories. Table 1 provides an overview of the structure of the dataset, serving as the foundation for the classification process. Calculate the prior probability by using Equation 5

$$\gamma(\text{positive}) = 1/3 \quad (6)$$

$$\gamma(\text{negative}) = 2/3 \quad (7)$$

Table 1: Sample dataset

Raining	ID	Review	Sentiment
Train set	1	Sweet food	Positive
	2	Not good As advertised	Negative
	3	Bad food	Negative
Test set	4	Bad food	Negative

Calculate conditional probabilities/maximum likelihood smoothing (Laplace) Naive Bayes estimate by using Equation 5

$$\gamma(\text{bad }|\text{positive} = \left(0 + \frac{1}{2+7} = 0.01235\right) \quad (8)$$

$$\gamma(\text{bad }|\text{negative} = \left(1 + \frac{1}{6+7} = 0.15385\right) \quad (9)$$

$$\gamma(\text{food }|\text{positive} = \left(1 + \frac{1}{2+7} = 0.2222\right) \quad (10)$$

$$\gamma(\text{food }\}\text{negative} = \left(0 + \frac{1}{6+7} = 0.0769\right) \quad (11)$$

Calculate posterior probability

$$\gamma(\text{positive} \mid 1\text{d}4) =$$

$$\frac{1}{3} * 0.02222 = 0.009129 \quad (12)$$

$$\gamma(\text{negative} \mid 1\text{d}4) =$$

$$2/3 * 0.222*0.0769=0.0113812 \quad (13)$$

$$\gamma(\text{negative} \mid 1\text{d}4) > \gamma(\text{positive}1\text{d}4) \quad (14)$$

$\gamma(\text{negative} \mid 1\text{d}4)$ is the maximum means probability of negative words in document 4 is maximum so document 4 is negative.

Performance evaluation

In this experiment, performance metrics are employed for the algorithm's accuracy analysis. The proposed system is evaluated using several accuracy measures which include: precision, recall, and F1-score.

1. **Precision: this** deals with the ability of the classifier not to tag a positive sample as

otherwise. How often the classifier is correct each time it predicts is defined as TP/(TP + FP)

2. **Recall;** deals with finding all positive instances by the classifier. It is defined as the sum of false negatives and the true positives ratio of true positives for each class. TP/ (TP + FN)

3. **F1-score:** It is the average mean of the two values which we have i.e. Precision and Recall. (To measure the accuracy of classifier for each class over others) as: (2∗precision-recall) /precision+recall.

RESULTS AND DISCUSSION

The dataset used for the experiments contains reviews about perishable and non-perishable food products from Amazon's web page with labels; Positive / Negative / SlightlyPos / Slightly Neg / Neutral). The sample dataset is shown in Table 2.

Table 2. Dataset**.**

ID	Review	Sentiment
1	Good quakity dog food	Positive
12	Not as advertised	Negative
23	"Delight" says it all	Slightly positive
34	Cough medicine	Neutral
45	Great taffy	Slightly positive
56	Nice taffy	Slightly positive
67	Great! Just as good as the expensive brands!	Positive
78	Wonderful, tasty taffy	Positive
89	Yay barley	Positive
910	Healthy dog food	Positive
1011	The best hot sauce in the world	Positive
1112	My cats LOVE this !diet! better than theirs	Positive
1213	My cats are not fans of the new food	Negative
1314	Fresh and greasy !	Slightly positive
1415	Strawbwrry Twizzlers-yummy	Positive

Table 3. Description of dataset

Name	**Variable type**	**Variable Description**
ID	Input	Unique ID of watch review
Review	Input	Comments about food products from social media pages
Sentiment	Output	The label associated with each review

The description of the dataset is given in Table 3

Experimental results

The experimental results for the two classifiers are presented in the form of confusion matrices, showcasing the counts of true positives, false negatives, true negatives, and false positives. These matrices offer a comprehensive view of the performance of each classifier, as outlined in Table 4. The True Positives (TP): tested for Positive & Review is actually positive. The True Negatives (TN): tested for Negative & Review is actually negative. The False Positives (FP): tested for Positive & Review is not (otherwise known as "Type I error."). The False Negatives (FN): tested

for Negative & Review is not. (Otherwise known as "Type II error.")

Table 4. Structure of confusion matrix

	Predicted class	
Actual class	True Neg. (TN)	False Pos. (FP)
	False Neg. (FN)	True Pos. (TP)

Results for Naïve Bayes' classifier

The Naïve Bayes algorithm was employed to classify the polarity of documents within the dataset. This algorithm categorizes reviews as either positive, slightly negative, slightly positive, neutral, or negative. Upon testing one of the reviews from the dataset, the outcome revealed its polarity classification. Table 5 displays the experimental results, indicating that 79,658 correct samples were identified out of 113,683 reviews using the Naïve Bayes classifier, as determined from the confusion matrix.

Table 5. Experimental result of Naïve Bayes' classifier

Total reviews	113,683
Classifier	Naive Bayes
Correct sample	79,658
Incorrect sample	34,025

The representations in Table 5 are given as follows:

correct samples = Summation of all TP values and
Incorrect samples = Summation of all FN and FP
Out of the total 113,683 reviews, 79,658 were correctly classified while 34,025 were incorrectly classified. The Naïve Bayes classifier demonstrated a higher number of correct classifications compared to the incorrect ones. The results of the confusion matrix of the Naïve Bayes classifier are given in Figure 3. Figure 4 shows the bar chart depicting output from the Naïve Bayes' classifier

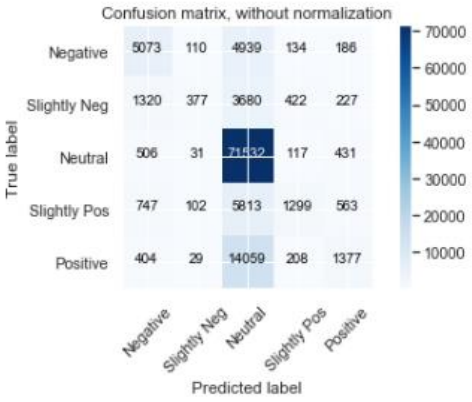

Figure 3. Confusion matrix of naïve bayes.

The implication of Figure 3 is that for the different data samples, the respective values for the positive variable are close to the actual values. This signifies that the model is accurate.

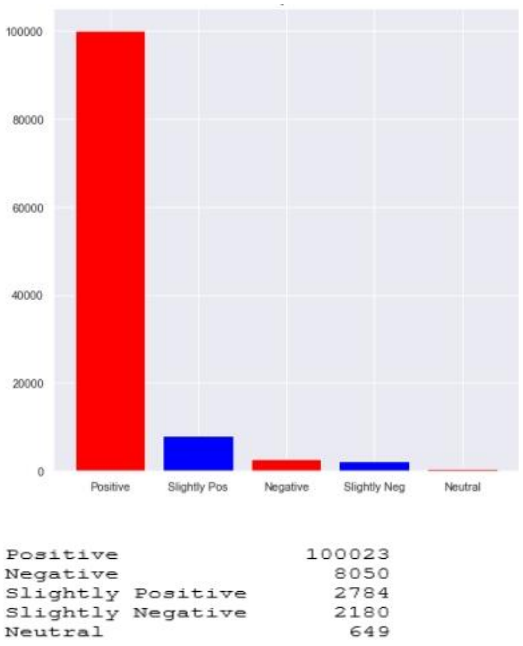

Figure 4. Bar chart depicting output from Naïve Bayes' classifier.

Results for Random Forest Classifier

The random forest algorithm was employed to classify the polarity of documents within the dataset. This algorithm categorizes reviews as positive, slightly negative, slightly positive, neutral, or negative. Upon testing one of the reviews from the dataset, the outcome revealed its

polarity classification. Table 6 displays the experimental results, indicating that 86,898 correct samples were identified out of 113,683 reviews using the Random Forest algorithm, as derived from the confusion matrix presented in Figure 5. Furthermore, Figure 6 illustrates a pie chart representing the output from the random forest classifier.

Where correct samples = Summation of all TP values *and*
Incorrect samples = Summation of all FN and FP

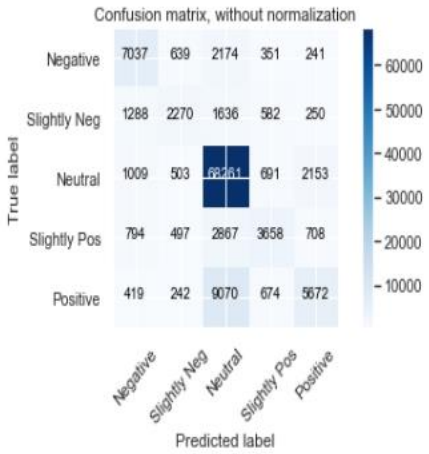

Figure 5. Confusion matrix of random forest.

Table 6. Experimental result.

Total reviews	113,683
Classifier	Random forest
Correct sample	86,898
Incorrect sample	26,788

Table 7: Naïve Bayes Classification Report

	Precision	Recall	F1 score
Negative	0.63	0.48	0.55
Neutral	0.58	0.06	0.11
Positive	0.72	0.99	0.83
Slightly neg.	0.59	0.15	0.24
Slightly pos.	0.51	0.09	0.15
Avg. Total	0.66	0.7	0.63

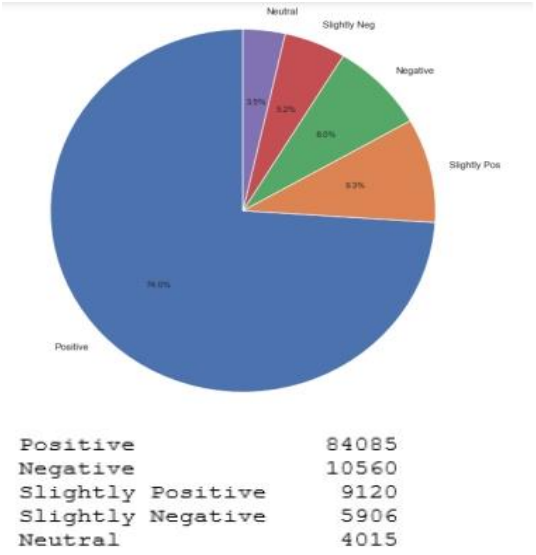

Figure 6. Pie chart depicting output.

Performance evaluation

Tables 7-8 are individual reports for the techniques utilized for performance evaluation.

Table 8: Random Forest Classification Report

	Precision	Recall	F1 score
Negative	0.67	0.66	0.66
Neutral	0.55	0.37	0.44
Positive	0.81	0.94	0.87
Slightly neg.	0.61	0.43	0.51
Slightly pos.	0.63	0.36	0.45
Avg. Total	0.74	0.77	0.74

Discussion

At the end of the experimental analysis, the result of Table 6 is obtained with a 70.01% accuracy on test data. Table 7 has 76.5%; therefore, the best accuracy was given by Table 7. The percentage accuracy of various classifiers is given in Table 9. Accuracy is calculated by:

$$\frac{Number\ of\ correct\ samples}{Total\ number\ of\ samples} \times 100 \qquad (6)$$

The implication of results obtained from Equation (6) is that the model produces more correct

samples than incorrect samples for the text classifiers.

Table 9. Percentage accuracy of various classifiers

Dataset	Classifier	Performance accuracy of classifier
Product review	Naïve bayes	70.01%
	Random forest	76.50%

CONCLUSIONS

For many large and mid-sized companies, understanding customer sentiments and opinions regarding their products and services is crucial due to the significant impact these sentiments can have on the company's financial performance. In this study, experimental analysis was carried out on a dataset comprising product reviews. Both the Naive Bayes classifier and the Random Forest classifier were utilized to train the dataset. It was observed that the Random Forest classifier outperformed the Naive Bayes classifier. Going forward, it is recommended to explore the development of a mobile application or a user-friendly graphical interface. Such tools would enable individuals without programming skills to easily assess and understand their customers' sentiments towards their products. This approach would facilitate broader accessibility and utilization of sentiment analysis tools, empowering companies to make informed decisions based on customer feedback.

REFERENCES

[1] Rejeb, A., Rejeb, K., Appolloni, A., Treiblmaier, H., & Iranmanesh, M. (2024). Exploring the impact of ChatGPT on education: A web mining and machine learning approach. The International Journal of Management Education, 22(1), 100932.

[2] Choe, S., Mehta, S. V., Ahn, H., Neiswanger, W., Xie, P., Strubell, E., & Xing, E. (2024). Making scalable meta learning practical. Advances in neural information processing systems, 36.

[3] Abubakr, M., Rady, M., Badran, K., & Mahfouz, S. Y. (2024). Application of deep learning in damage classification of reinforced concrete bridges. Ain Shams engineering journal, 15(1), 102297.

[4] Mupaikwa, E. (2025). The Application of Artificial Intelligence and Machine Learning in Academic Libraries. In Encyclopedia of Information Science and Technology, Sixth Edition (pp. 1-18). IGI Global.

[5] BÜYÜKKEÇECİ, M., & OKUR, M. C. (2024). A Comprehensive Review of Feature Selection and Feature Selection Stability in Machine Learning. Gazi University Journal of Science, 1-1.

[6] Valtonen, L., Mäkinen, S. J., & Kirjavainen, J. (2024). Advancing reproducibility and accountability of unsupervised machine learning in text mining: Importance of transparency in reporting preprocessing and algorithm selection. Organizational Research Methods, 27(1), 88-113.

[7] Shamshiri, A., Ryu, K. R., & Park, J. Y. (2024). Text mining and natural language processing in construction. Automation in Construction, 158, 105200.

[8] Duan, H. K., Vasarhelyi, M. A., Codesso, M., & Alzamil, Z. (2023). Enhancing the government accounting information systems using social media information: An application of text mining and machine learning. International Journal of Accounting Information Systems, 48, 100600.

[9] Umer, M., Imtiaz, Z., Ahmad, M., Nappi, M., Medaglia, C., Choi, G. S., & Mehmood, A. (2023). Impact of convolutional neural

network and FastText embedding on text classification. Multimedia Tools and Applications, 82(4), 5569-5585.

[10] Pal, S., Biswas, B., Gupta, R., Kumar, A., & Gupta, S. (2023). Exploring the factors that affect user experience in mobile-health applications: A text-mining and machine-learning approach. Journal of Business Research, 156, 113484.

[11] Kariri, E., Louati, H., Louati, A., & Masmoudi, F. (2023). Exploring the advancements and future research directions of artificial neural networks: a text mining approach. Applied Sciences, 13(5), 3186.

[12] Abdusalomovna, T. D. (2023). TEXT MINING. European Journal of Interdisciplinary Research and Development, 13, 284-289.

[13] Spiteri, G., Fielding, J., Diercke, M., Campese, C., Enouf, V., Gaymard, A., Bella, A., Sognamiglio, P., Moros, M.J.S., Riutort, A.N. and Demina, Y.V., 2020. First cases of coronavirus disease 2019 (COVID-19) in the WHO European Region, 24 January to 21 February 2020. Eurosurveillance, 25(9), p.2000178

[14] Spiteri, G., Fielding, J., Diercke, M., Campese, C., Enouf, V., Gaymard, A., Bella, A., Sognamiglio, P., Moros, M.J.S., Riutort, A.N. and Demina, Y.V., 2020. First cases of coronavirus disease 2019 (COVID-19) in the WHO European Region, 24 January to 21 February 2020. Euro surveillance, 25(9), p.2000178