DOI: http://dx.doi.org/10.26483/ijarcs.v14i6.7029

ISSN No. 0976-5697

Volume 14, No. 6, November - December 2023

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

E-COMMERCE PRODUCT RATING BASED ON CUSTOMER MINING FOR COMMENTS USING MACHINE LEARNING TECHNIQUE

Salah Zaher

TAIBA University, College of Business Administration,
Department of Management Information System,
Saudi Arabia,Cairo University, Egypt

Abstract The E-commerce industry heavily relies on customer reviews to gauge product performance. In this paper, we leverage machine-learning techniques to analyze customer comments on a specific product. Our system employs data sets from Kaggle that include popular cell phones from around the world, classifying reviews as positive or negative. This approach helps sellers and company owners selling products online gain insight into customer satisfaction or dissatisfaction with their products. By analyzing customer feedback, businesses can improve their products and boost profits. To test our system, we used the Google Colab environment and experimented with three different algorithms: naïve bias, decision tree, and forest decision tree. The results indicated that the naïve bias algorithm had the highest accuracy (91.3%), precision (95.5%), recall (86.6%), and F-score (91%).

Keywords- E-commerce, Machine learning, Kaggle, Google Colab.

1. INTRODUCTION

The use of web technologies has become more popular in recent years. As a result, most individuals use the online application to voice their thoughts on buying and selling things. User-review sites are fast growing in popularity. On the internet, a customer may resort to writing a review about the goods he wishes to purchase. Hundreds of reviews are left on certain products. For both buyers and businesses, these opinions and reviews have become a valuable resource. Companies can use these opinions and reviews to improve and develop the items they offer. Because certain items have more significant characteristics than others and have a greater impact on a customer's purchase choice, the important aspects must be recognized [1]. A large number of users read the comments because they value what others have to say. These comments have an impact on the product; if the comments are favorable, the overall impression is positive; if the impression is negative and suspicious, the product suffers and the business or institution's reputation suffers. Some product remarks may be inaccurate, affecting the product's reputation. As a result, to grow earnings, companies have to offer positive and honest feedback. There are problems with evaluations and solutions to those problems. Consumers have reason to suspect comments and opinions, thus authenticity must be detected and authenticated such that fraudulent remarks are less likely. The Kaggle website provides a dataset containing comments from Amazon customers regarding cell phones from ten brands: Nokia, Apple, Samsung, OnePlus, Sony, ASUS, Google, HUAWEI, Motorola, and Xiaomi. [2]

2. DATA PREPROCESSING:

To effectively analyze customer comments, it is crucial to efficiently process the data in a useful manner. One way to do this is through data mining techniques that transform raw data into a more organized and manageable format. The initial steps involve converting all letters to lowercase, eliminating punctuation marks, and removing stop words such as "the," "is," and "a." Additionally, normalization should be applied to address common misspellings and abbreviations in comments. This includes converting text to a standard form, such as changing "gooooooood" to "good." By implementing these steps, the data becomes easier to analyze, enabling the identification of positive and negative opinions from customers. Upon completion of data processing, a dataset comprised exclusively of fundamental words is obtained, amenable to analysis and mining. This analytical process entails the identification of the most frequently occurring words, both positive and negative; using a variety of data mining algorithms.Fig2 shows a sample of the data set. Fig3Average rating count for positive and negative reviews

3. RELATED WORK

Customer satisfaction is crucial in marketing and research, especially regarding consumer behavior. Excellent service in hotels results in positive word-of-mouth transmission [3].

In their study, Heng et al [4] aimed to provide valuable insights into the factors that influence consumers' choices of food products when shopping online, given the increasing popularity of online food and grocery shopping. To achieve this, they analyzed Amazon's customer comments data using the R programming language. They cleaned and processed the comments texts to create a corpus of relevant comments by removing irrelevant and infrequent terms, making all terms lowercase, and eliminating numbers and special characters. They also combined words with the same root into a single term.

Using the Late Dirichlet Allocation (LDA) algorithm, they identified the comments with a rating of 4 and above as good comments and generated a dummy variable called "Good." These comments accounted for nearly 85% of the total comments, indicating that consumers generally had a positive experience with the coffee products available on Amazon.com.

A study [5] was conducted to gather data from Twitter and Flexstar, to extract movie ratings from various sources. The data underwent data cleaning, and the text was converted to lowercase using Rapid Miner, Weka, and RStudio to help with data manipulation and comprehension. The paper utilized a dictionary-based algorithm for sentiment analysis, with the polarity of reviews being a crucial factor in determining the overall sentiment. The study not only focused on the sentiment of reviews but also predicted the movie rating using a machine-learning class. The data was trained based on features such as popularity and positive and negative polarity percentages, after which the rating was assigned to a particular movie. The Naive Bayes algorithm resulted in the highest accuracy of 54.1%, with 73% precision, 66%, recall, and an F-score of 61.6%, while the decision tree yielded 44.26% accuracy, 46% precision, 56% recall, and F-score 0f of 62.6.

4. PROPOSED METHODS

In this section, we shall discuss two types of machine learning classifiers: Naïve Bayes , Decision Tree and Random forest. The topic of classification has been extensively researched in the fields of database management, data mining, and information retrieval. Classification involves identifying the class value of a set of training records (D = {X1,...,XN}) which are labeled with discrete values indexed by {1 ...k}. A classification model is constructed using the training data to link the features of a record to one of the available class labels. [6].

4 .1 Naïve Bias Classifier

The widely used Naïve Bias algorithm is a supervised technique that excels in classification tasks. It offers exceptional scalability, making it well-suited for two-layer or multiple-layer classification. This algorithm is driven by conditional probability, making it highly effective for analyzing textual data and performing sentiment analysis on customer feedback, including both positive and negative comments. Additionally, a speedy solution can deliver real-time predictions. The fundamental Naïve Bayes assumption is that each feature makes an Independent equal contribution to the outcome the formula is:

P (A|B) = P (A AND B) / P (B)

A naive Bayes model can be seen as a collection of unigram language models that are specific to each class. Each class's model creates a unigram language model. The likelihood features of the naive Bayes model assign a probability to each word, P(word|c), and subsequently assign a probability to each sentence as well: P(s|c) = Y i∈positions P(wi |c)[7].

4.2 Decision Tree Classifier

Experts in statistics, machine learning, pattern recognition, and data mining have discovered that decision trees are an effective method for representing classifiers. This involves creating a decision tree based on existing data. A decision tree is a classifier expressed as a recursive partition of the instance space. The decision tree consists of nodes that form a rooted tree, meaning it is a directed tree with a node called the "root" that has no incoming edges. All other nodes have only one incoming edge. A node with outgoing edges is referred to as an internal or test node, while all other nodes are called leaves or terminal nodes. In a decision tree, each internal node divides the instance space into two or more sub-spaces according to a certain discrete function of the input attribute values. Each test considers a single attribute, and the instance space is partitioned based on the attribute's value. For numeric attributes, the condition refers to a range. Each leaf is assigned to one class representing the most appropriate target value. Alternatively, the leaf may hold a probability vector indicating the probability of the target attribute having a certain value. Instances are classified by navigating them from the root of the tree down to a leaf, according to the outcome of the tests along the path. Internal nodes are represented as circles, whereas leaves are denoted as triangles. The decision tree incorporates both nominal and numeric attributes. With this classifier, an analyst can predict the response of a potential customer by sorting it down the tree and understanding the behavioral characteristics of the entire potential customer population regarding direct mailing. Each node is labeled with the attribute it tests, and its branches are labeled with their corresponding values [8].

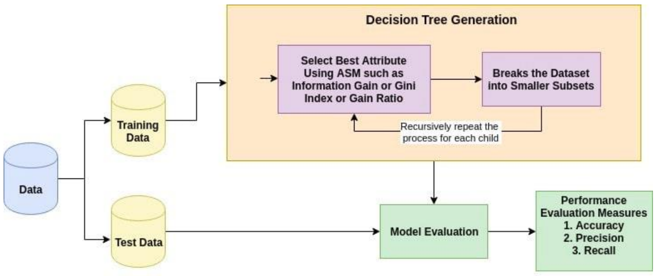
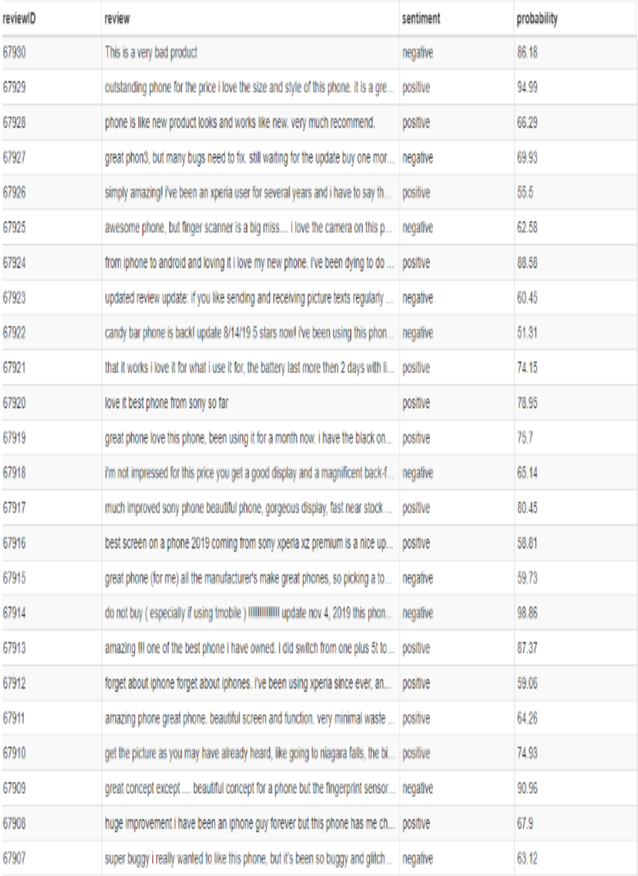

Decision Tree Generation
Select Best Attribute Using ASM such as Information Gain or Gini Index or Gain Ratio
Breaks the Dataset into Smaller Subsets
Recursively repeat the process for each child
Training Data
Data
Test Data
Model Evaluation
Performance Evaluation Measures
1. Accuracy
2. Precision
3. Recall

Fig 1 Block diagram of the decision tree

4.3 Random Forest Classifier

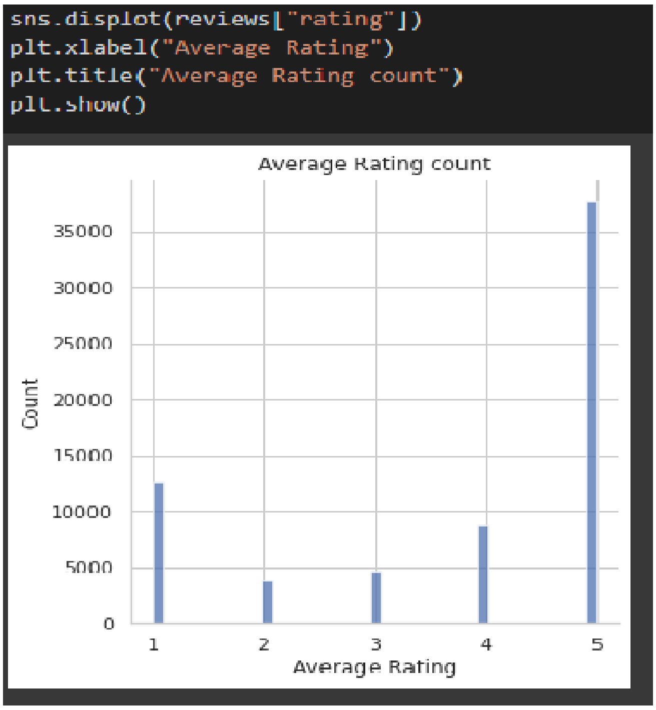
To overcome the limitations of relying on a single prediction model, a researcher has developed a new ensemble method called Random Forest. This approach involves training multiple models using subsets of the same dataset. By combining the predictions of these models, the ensemble is able to achieve higher accuracy than a single model. The idea of using an ensemble approach dates back to the 1970s when researchers began combining multiple models to improve their forecasting accuracy [9].

Random Forest utilizes decision trees as base classifiers. It generates multiple decision trees with randomization in two ways: (1) random sampling of data for bootstrap samples, as is done in bagging, and (2) random selection of input features for generating individual base decision trees. The strength of individual decision tree classifiers and the correlation among base trees are key issues that determine the generalization error of a Random Forest classifier. The accuracy of Random Forest classifier has been found to be on par with existing ensemble techniques like bagging and boosting. According to Breiman, Random Forest runs efficiently on large databases, can handle thousands of input variables without variable deletion, provides estimates of important variables, generates an internal unbiased estimate of generalization error as forest growing progresses, has an effective method for estimating missing data, maintains accuracy when a large proportion of data is missing, and has methods for balancing class error in class population unbalanced data sets [10]. A group of separately trained predictors, like neural networks or decision trees, that work together to classify new data instances is called an ensemble. Research has demonstrated that ensembles are frequently more precise than any single classifier in the group [11], [12],. The concept of bootstrap samples involves generating multiple classifiers from an original training dataset of size N in order to create an ensemble. If m individual classifiers are desired, m different training sets are generated from the original dataset by sampling with replacement. These classifiers are independent of each other in bagging [13]. On the other hand, in boosting[14], weights are assigned to each sample from the training dataset. If m classifiers are to be generated, they are created sequentially, with one classifier generated in each iteration. To generate classifier Ci, the weights of training samples are updated based on the classification results of classifier Ci-1. The classifiers produced by boosting are dependent on each other.The concept of bootstrap samples involves generating multiple classifiers from an original training dataset of size N in order to create an ensemble. If m individual classifiers are desired, m different training sets are generated from the original dataset by sampling with replacement. These classifiers are independent of each other in bagging. On the other hand, in boosting, weights are assigned to each sample from the training dataset. If m classifiers are to be generated, they are created sequentially, with one classifier generated in each iteration. To generate classifier Ci, the weights of training samples are updated based on the classification results of classifier Ci-1. The classifiers produced by boosting are dependent on each other.



reviewID	review	sentiment	probability
67930	This is a very bad product	negative	86.18
67929	outstanding phone for the price i love the size and style of this phone. it is a gre...	positive	94.99
67928	phone is like new product looks and works like new. very much recommend.	positive	66.29
67927	great phon3, but many bugs need to fix. still waiting for the update buy one mor...	negative	69.93
67926	simply amazing! i've been an xperia user for several years and i have to say th...	positive	55.5
67925	awesome phone, but finger scanner is a big miss.... i love the camera on this p...	negative	62.58
67924	from iphone to android and loving it i love my new phone. i've been dying to do ...	positive	88.58
67923	updated review update: if you like sending and receiving picture texts regularly ...	negative	60.45
67922	candy bar phone is back! update 8/14/19 5 stars now! i've been using this phon...	negative	51.31
67921	that it works i love it for what i use it for, the battery last more then 2 days with li...	positive	74.15
67920	love it best phone from sony so far	positive	78.95
67919	great phone love this phone, been using it for a month now. i have the black on...	positive	75.7
67918	i'm not impressed for this price you get a good display and a magnificent back-f...	negative	65.14
67917	much improved sony phone beautiful phone, gorgeous display, fast near stock ...	positive	80.45
67916	best screen on a phone 2019 coming from sony xperia xz premium is a nice up...	positive	58.81
67915	great phone (for me) all the manufacturer's make great phones, so picking a to...	negative	59.73
67914	do not buy (especially if using tmobile) !!!!!!!!!!!! update nov 4, 2019 this phon...	negative	98.86
67913	amazing !!! one of the best phone i have owned. i did switch from one plus 5t to...	positive	87.37
67912	forget about iphone forget about iphones. i've been using xperia since ever, an...	positive	59.06
67911	amazing phone great phone. beautiful screen and function. very minimal waste ...	positive	64.26
67910	get the picture as you may have already heard, like going to niagara falls, the bi...	positive	74.93
67909	great concept except .... beautiful concept for a phone but the fingerprint sensor...	negative	90.96
67908	huge improvement i have been an iphone guy forever but this phone has me ch...	positive	67.9
67907	super buggy i really wanted to like this phone, but it's been so buggy and glitch...	negative	63.12

Figure 2: Sample of the data set



```
sns.displot(reviews["rating"])
plt.xlabel("Average Rating")
plt.title("Average Rating count")
plt.show()
```

Figure 3: Average rating count of more than 3 is a positive review otherwise it is a negative review

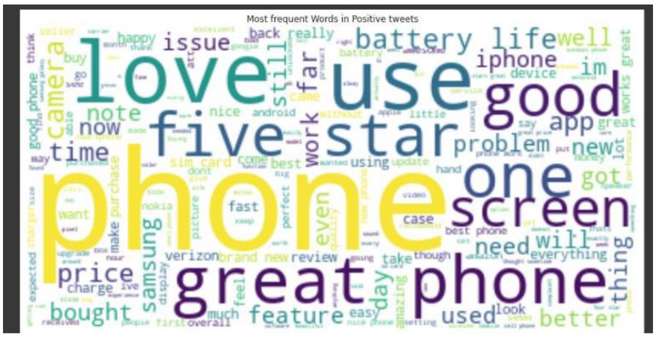


Figure 4: Positive sentiments word cloud generation

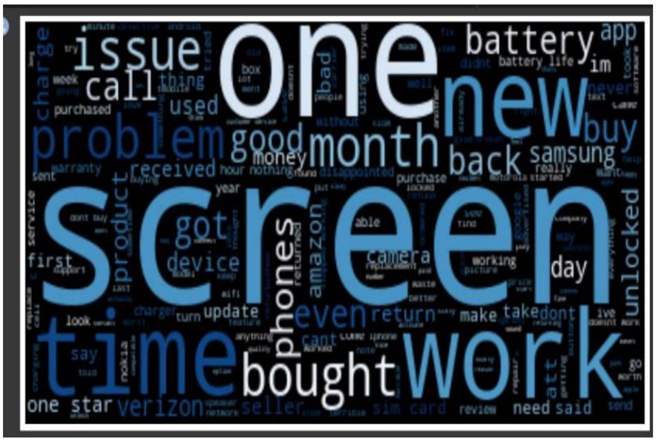


Figure 5: Negative
Sentiments word cloud generation

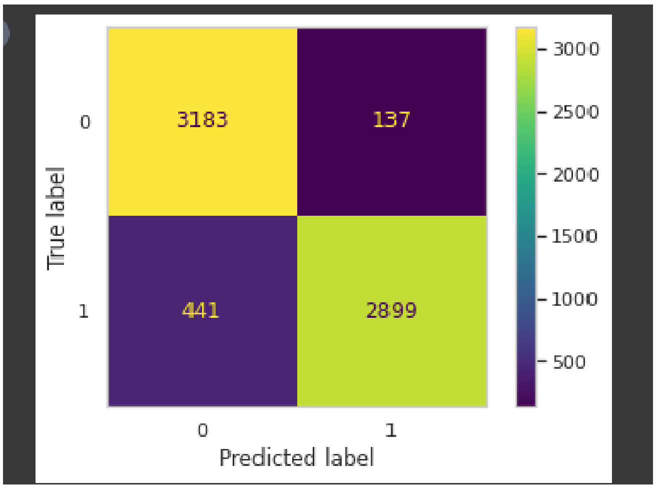


Figure 6: confusion matrix using Naïve Bias

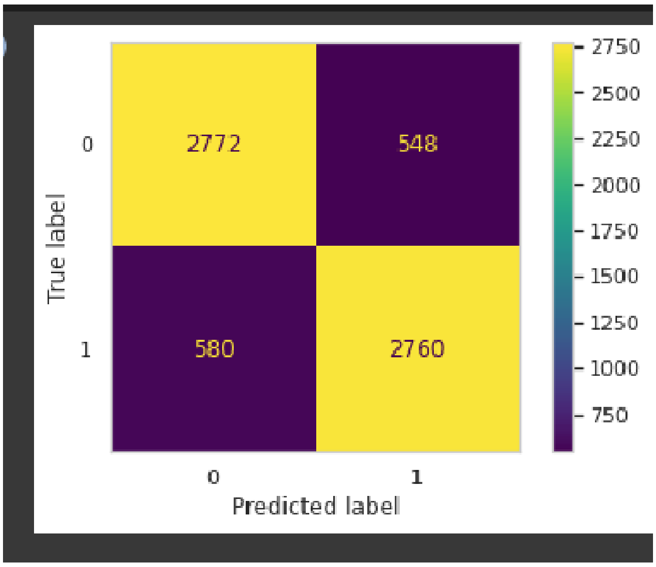


Figure 7: confusion matrix using Decision Tree

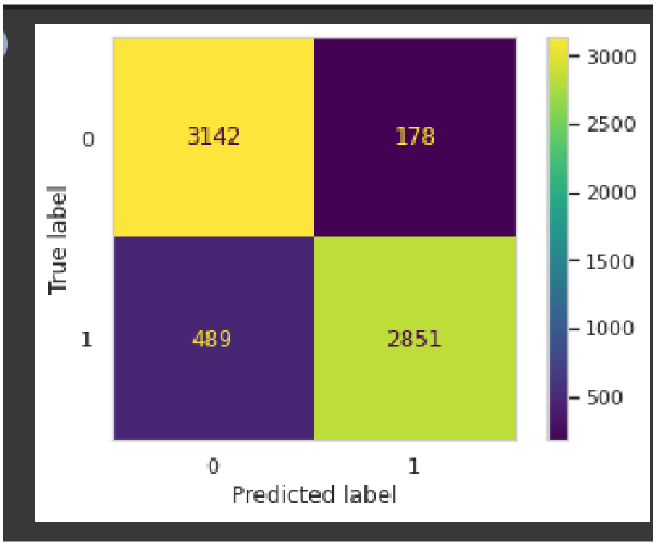


Figure 8: confusion matrix using Random Forest

5. Tools Used

- Google Colab environment
The Python environment on Google servers is used to run scripts for analysis, preprocessing, and machine learning libraries.

6. Methodolgy

It is crucial to fine-tune hyperparameters when training a Naïve Bayes model. The code for hyperparameter tuning is shown in Figure 9, and the results are displayed in Fig 10. The results of the three classifiers—the naive Bayes, decision trees, and random forests—will be compared to select the algorithm with the best ability to classify the user comments.

7. Confusion Matrix

The confusion matrix is a highly useful tool in machine learning when it comes to predictive analysis. It plays a crucial role in evaluating the performance of classification-based machine learning models. Essentially, the matrix provides a summary of the correct and incorrect predictions made by a classifier or classification model for binary classification tasks. The matrix is an N x N grid used to assess the performance of a classification model, where N is the number of target classes. One can determine the accuracy of the model by analyzing the diagonal values of the confusion matrix, which represents the number of accurate classifications. The confusion matrix in fig6.can is divided into four regions [15]:

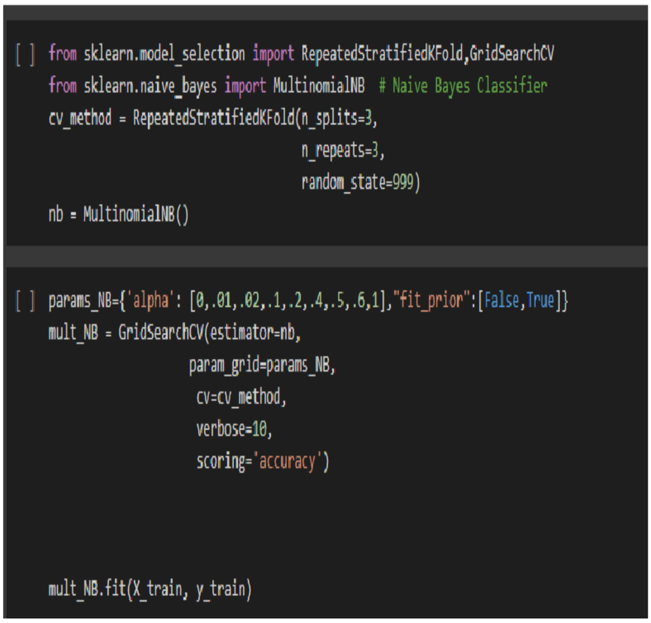
$$presion = \frac{TP}{TP + FP}$$

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

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

$$FScore = \frac{2 * precision * Recall}{Precision + Recall}$$

TP = the positive review of real data
classified as the positive review
TN= the negative review of real data
classified as the negative review
FP= the negative review of real data
classified as the positive review
FN= the positive review of real data
classified as the negative review



```
[ ] from sklearn.model_selection import RepeatedStratifiedKFold,GridSearchCV
    from sklearn.naive_bayes import MultinomialNB  # Naive Bayes Classifier
    cv_method = RepeatedStratifiedKFold(n_splits=3,
                                        n_repeats=3,
                                        random_state=999)

    nb = MultinomialNB()
```

```
[ ] params_NB={'alpha': [0,.01,.02,.1,.2,.4,.5,.6,1],"fit_prior":[False,True]}
    mult_NB = GridSearchCV(estimator=nb,
                           param_grid=params_NB,
                            cv=cv_method,
                            verbose=10,
                            scoring='accuracy')



    mult_NB.fit(X_train, y_train)
```

Fig.9 Hyperparmeter Tuning Naïve Bayes

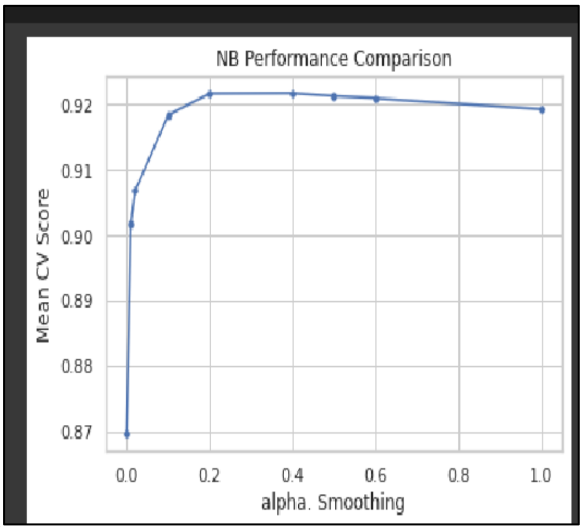


Fig 10 alpha smoothing of NB algorithm

8. RESULTS

Figures 4 and 5 display word clouds of positive and negative sentiments, respectively. Confusion matrices for Naïve Bias, Decision Tree, and Random Forest can be found in Figures 6, 7, and 8. Table 1 shows the accuracy, precision, Recall and Fscore of the three algorithms, which was calculated based on the confusion matrix..

TABLE 1: METRIC RESULTS FOR THE THREE ALGORITHMS

	Naïve Bias	Decision Tree	Forest Decision tree
Accuracy	91.32%	83.06%	89.98%
Precision	95.49%	83.43%	94.12%
Recall	86.80%	82.63%	85.36%
F Score	91%	83%	90%

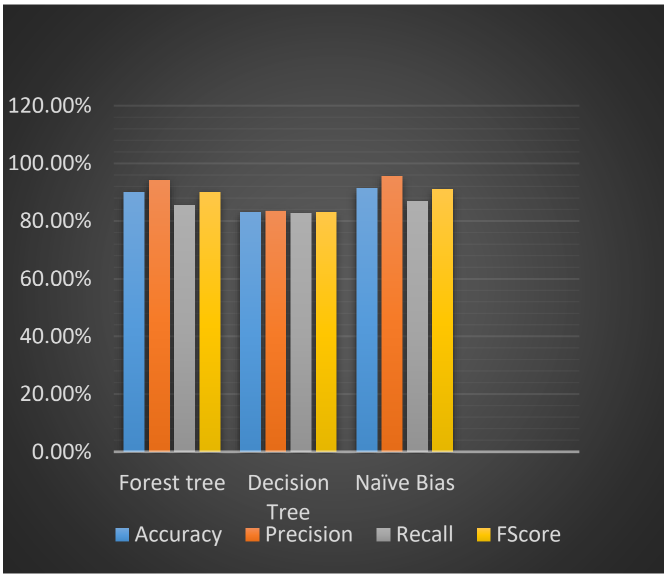


Fig 11 Comparison of the three algorithms

Figure 11 compares the accuracy, precision, recall, and F-score of three algorithms. The Naïve Bayes algorithm achieved the highest score across all metrics.

9 . CONCLUSION

Based on extensive testing, it has been established that the Naïve Bayes classifier is an excellent choice for sentiment analysis, achieving an impressive accuracy rate of 91.3%, a precision score of 95.5%, recall rate of 86.6% and F score 91%. Further analysis is recommended to compare the Naïve Bayes algorithm with other algorithms not examined in this study. Additionally, investigating other parameters that may affect the performance of the Naïve Bayes algorithm could lead to further improvements in its overall performance.

10. COMPETING INTEREST AND FUNDINGS

I hereby state that I do not have any competing interests, as defined by Springer or any other organization that could potentially influence the results or discussion presented in this paper. Furthermore, I confirm that I am not receiving any funds from any individual or organization.

11. REFERENCES

[1]Anggadwita G &Martini E "Digital Economy for customer benefits and business fairness 2020 Taylor & Francis Group, London, UK ISBN:978-0-367-47722-6(Hbk)

[2]Nibras, G. (2019, December 26). Amazon cell phones reviews. Kaggle. Retrieved November 28, 2021, from https://www.kaggle.com/grikomsn/amazon-cell-phones-reviews.

[3] Zainol, N., Rozali, A., & Nordin, N. (2016, Jan). "The Influence of Customer Satisfaction Towards Positive Word-of-Mouth in Hospitality Industry". SSRN Electronic Journal Published by Elsevier BV

[4] Heng, N., Gao, Z., jiang,Y. & Chen, X. (2018, May). "Exploring hidden factors behind online food shopping from Amazon reviews: A topic mining approach". Journal of Retailing and Consumer Services

[5] Singh, V., Saxena, P., & Siddharth Singh and S Rajendran. (2017, May 11). Opinion mining
and analysis of Movie Reviews. SRS Journal. Retrieved November 6, 2021, from https://indjst.org/articles/opinion-mining-and-analysis-of-movie-reviews.

[6] Rathor, A. S., Agarwal, A., & Dimri, P. (2018, June 8). Comparative study of machine learning approaches for amazon reviews. Procedia Computer Science 132:1552-1561.

[7] Aggarwal, C. C. and C. Zhai. 2012. A survey of text classification algorithms. In C. C. Aggarwal and C. Zhai, editors, Mining text data, pages 163–222. Springer.

[8] Lior Rokach and Oded MaimonOded Maimon. The Data Mining and Knowledge Discovery Handbook Springer US, January 2005 ISBN 978-0-387-24435-8, 978-0-387-25465-4.

[9] BBiosc.Biotech.Res.Comm. Special Issue Vol 13 No 14 (2020) Pp-245-248 A Detailed Review on Decision Tree and Random Forests Bhushan Talekar1 and Sachin Agrawal2

[10] Brieman L, Random Forests, Machine Learning, 45, 5-32, (2001) © 2001 Kluwer Academic Publishers. Manufactured in The Netherlands.

[9] Speech and Language Processing. Daniel Jurafsky & James H. Martin. Copyright © 2023.

[11] Krogh A, Vedelsby J, Neural Network Ensembles,
Cross Validation, and Active Learning, Advances
in Neural Information Processing Systems Vol 7,
MIT Press , 231-238, (1995)

[12]Opitz D, Maclin R, Popular Ensemble Methods: An
Empirical Study, Journal of Artificial Intelligence
11, 169-198, (1999)

[13] Breiman L, Bagging Predictors , Technical report
No 421, (1994)

[14] Robert E Schapire, The Boosting Approach to
Machine Learning an Overview, Nonlinear Estimation and Classification, Springer, 2003

[15] Zohreh Karimi October 2021"Confusion Matrix" https://www.researchgate.net/publication/355096788