***Type: xxx***

**Fake Product Review Detection using Machine Learning**

**Md. Iqbal Hossain1, Fariha Ahmed Raha1, Mohammad Monirujjaman Kha****n1\***

1Department of Electrical and Computer Engineering, North South University, Bashundhara, Dhaka-1229, Bangladesh

\*Corresponding Author: Mohammad Monirujjaman Khan. Email: monirujjaman.khan@northsouth.edu

Received: XX Month 202X; Accepted: XX Month 202X

**Abstract:** Online reviews play a crucial role in determining whether a product will be sold on e-commerce websites or applications such as the Amazon, Shwapno, Daraz, and others. Because so many people rely on internet evaluations, unethical actors may fabricate reviews in order to artificially boost or devalue items and services. The majority of individuals are looking for accurate information regarding an online product. Before investing their money in a specific product, they can read many reviews on the website. In this case, they were unable to tell whether the item was real. In general, certain reports on websites are useful, and the company's technical staff add them to help publicize the product. These individuals are members of the media and social organization teams who provide positive ratings for their own company. Because of the falsification in the website reviews, online buyers were unable to detect the fraudulent product. To detect false product reviews, this research provides a semi-supervised machine learning approach. Furthermore, many engineering features are used in this work to extract diverse reviewer behaviors. This study examines the outcomes of numerous experiments on a real food review dataset of restaurant reviews with attributes collected from user behavior. The effectiveness of two classifiers, Naïve Bayes (NB) and Random Forest, is evaluated to compare. In terms of f-score, the results indicate that Random Forest surpasses another classifier, with the best f-score of 98 %. More behavioral features must be held in order to improve the efficacy of the offered fake review detecting algorithm. In addition, the data reveals that taking into account the reviewers' behavioral characteristics raises the f-score by 55.5%. In the current technique, not all reviewers' behavioral characteristics have been considered. Other low-level features such as frequent time or date dependency, the reviewer's timing for giving a review, and how common it is to deliver favorable or poor reviews will be added further in order to improve the efficacy of the offered fake review detecting algorithm.

**Keywords:** Machine Learning; Fake reviews; Semi-supervised; Naïve Bayes; Random Forest; F-Score

**1 Introduction**

People are now buying products online, owing to the World Wide Web's rising interest. As more individuals become accustomed to the internet and find it easy to use, a rising number of customers are posting their reviews, ratings, and comments online. This practice of submitting reviews is also proven to be highly beneficial to other people who make purchases over the internet. As a result, the number of reviews has risen, as has their significance on product sales. Hundreds of reviews are written for popular products, and these reviews can make or break their sales. Customers can offer comments on e-commerce sites in general. The fact that these evaluations exist can be exploited as a data source. Companies, for example, might utilize it to make product or service design decisions. Unfortunately, some parties have sought to devalue the review by generating false reviews, both to increase the popularity of the product and to discredit it. They post their ideas on the internet. It is a common human habit to conduct a poll on a product before purchasing it. Customers may compare brands and decide on a product based on customer reviews. These internet reviews have the power to influence a customer's purchasing decision. For example, if a customer gives extremely negative comments on an Apple review website about a mobile phone, such as the iPhone 8, it is due to poor service. This review will give potential buyers a negative picture of the iPhone 8 and harm its business. As a result, in order to avert financial losses as a result of such genuine reviews, Apple Inc.'s owner. To improve its reputation, the company could hire some people to write false reviews. False reviews may lead a buyer to acquire a low-quality product, while untruthful reviews may taint good services or high-quality products. Spammers are those that write malicious reviews in order to purposely mislead consumers or opinion analysis systems, and false reviews are referred to as spam reviews. Users will be able to choose the best product for their needs if these reviews are reliable. The user may be misled if the reviews are slanted or fraudulent.

Before making an online purchase, a lot of people read internet reviews. However, because reviews can be corrupted or manufactured for financial benefit or profit, any decision based on online reviews should be taken with caution. Furthermore, business owners may offer incentives to anyone who posts positive evaluations about their products or services, or they may pay someone to write negative evaluations about their competitors' products or services. Because of the importance of reviews, these fraudulent reviews are termed review spam and can have a significant impact on the online economy. This encourages the development of a system that uses the content and rating properties of a review to detect phony product reviews. Data mining techniques will be used to determine the honest value and identify a fake review.

Machine learning is an area of AI that includes using computers to simulate human learning. This enables computers to recognize and learn knowledge from the current world, allowing them to improve their performance on certain tasks as a result of new information. Semi-supervised learning and reinforcement learning methodologies are used in Machine learning algorithms. Machine learning enables consumers to customize their needs and become aware of the most popular products on the market. In today's world, "what other people think and how they think" has always been a significant source of information for decision-making. No Longer before the internet existed or became as ubiquitous as it is today, people still polled each other about goods and products or relied on surveys to make decisions.

Machine learning is a significant technological innovation that is used in a range of crucial applications. Machine learning's primary strength is in assisting machines in learning and improving automatically based on prior experience. Machine learning algorithms are classified as supervised, semi-supervised, or unsupervised. Both input and output data are presented in the unexpected technique, and the training data must be labeled and categorized. The main purpose of the unsupervised learning technique is to determine input data clustering or classification that best fits the input data without any categorization or labeling. As a result, every dataset in unsupervised learning is unlabeled, and the approach's job is to label them. Finally, some data is tagged in the semi-supervised strategy, but the majority is not. This section provides an overview of supervised learning algorithms, which are the subject of this research.

To identify false product reviews, this research provides a semi-supervised machine learning approach. Furthermore, to extract distinct reviewer behaviors, this study uses multiple engineering elements. This research examines the results of many experiments employing attributes obtained from user behavior on a real-world dataset of restaurant assessments. This study proposes a semi-supervised machine learning strategy for detecting fake product reviews. Furthermore, many engineering features are used in this work to extract diverse reviewer behaviors. The results of multiple trials on a real food review dataset of restaurant reviews are matched to attributes gathered from user behavior in this study. The results of two classifiers, Naive Bayes (NB) and Random Forest (RF), are compared and contrasted. One of the programs will aid customers in selecting the best product for their needs. The algorithm will conduct an analysis, and if any phony reviews are identified on a regular basis from a specific IP address, the admin user will have the option to restrict that IP address. The user will also receive a message with the blacklisted IP address. It keeps track of any fake product reviews in this way. In addition, the consumer can be sure that the products are available on that app, as well as review them. Other research has employed the SVM classification process to detect bogus reviews using IP addresses. This application aids people in obtaining accurate product reviews on the internet. The accuracy improved by 98.79 %, and the F1 score improved by 10%. [1] In [2,] they employed sentiment analysis to discover bogus and spam reviews and then removed them using the J48 Algorithm and the Naïve Bayes Algorithm. So, the proposed system will save time and effort by assisting consumers and businesses in quickly identifying spam from various perspectives and assist them in acquiring their important products from a trustworthy site. It aims to detect false reviews from a collection of product reviews in [3], which involves simulating fake reviews with different kinds of opinion spam review features, creating a training set, then characterizing them using Naïve Bayes classification and ensemble classification models like Random Forest to test the model.

In [4], it shows that employing intelligent learning approaches, the suggested study obtained an accuracy of 87 % in detecting false reviews written in English, which is higher than previous systems' accuracy. They worked on this article with the goal of removing fraudulent reviews from the original reviews, which is a pressing issue. The major goal of our work on [5] is to develop a system that can detect spam and redundant reviews and filter them out so that users can learn more about the product. The purpose of this study is to improve customer satisfaction while also making online shopping more secure. Commentator Centric Approach—This strategy is based on analyst behavior—was utilized in the paper [6] for detecting reviews. This strategy considers client information as well as all surveys created with their assistance. Highlights now include age of the account, profile image, url duration, IP address, written audits by a single reviewer, and the maximum harsh ratings. The item centric approach is a technique for locating the majority of component facilities based on object-relevant data. Currently, the rank of an object, its value, and other factors are taken into account as highlighted. As a result, according to [7], the quantity of product reviews is rapidly increasing. Thousands of reviews may be found on various websites for the majority of products. Nowadays, each consumer can write his or her own opinion message or review, attracting the attention of individual and allowing companies to supply unworthy spam opinions in order to advertise or discredit certain specific products. Spam, invalid reviews and comments may not be limited by the recent system. As a result, there is a need to create a rational system that can simultaneously mine ideas and categorize them as spam or non-spam. Select a group of highly suspect reviewers for additional investigation by user evaluators using an internet-based spammer evaluation software specifically designed for user evaluation trials [8].

Finally, it has been demonstrated that the discovered scammers have a greater influence on ratings than the unhelpful reviewers. It employs the Naïve Bayes and Sentimental Analysis Algorithms. The findings suggest that proposed ranking and supervised approaches are excellent at detecting spammers, outperforming current baseline methods that rely solely on helpfulness votes. Behavioral aspects have been considered in the detection of bogus reviews in other studies. In Amazon reviews, some characteristics, like the average ratings and the ratio of reviews submitted by the reviewer, have been taken into account. Another study looked into the major effect of textual and behavioral problems on the detection of fraudulent reviews in the restaurant and hotel industry.

Some systems can detect false reviews posted by the optimization team by detecting the IP address. The user will log in with his user ID and password, browse a variety of products, and submit reviews for each of them. In addition, the user will receive authentic product reviews. And, while reviewing them, he must input the email address he is reviewing, which will be validated. If he publishes a bogus review, his ID will be disabled, and he will not be able to voice his thoughts again. [9] Another study found that by properly selecting features that can capture diverse aspects of valid comments in order to distinguish from spam comments made by spammers, it is possible to detect spam comments. [10-11]

First, a dataset is utilized to identify and distinguish between good and bad reviews using keywords commonly used in reviews, and then the user's input is chosen to detect spammed reviews. Following the selection, the product is checked for spam. [12-13] The J48 Algorithm is used to detect spam in reviews, and such reviews are discriminated against using a decision tree, and spam reviews are identified. Following the detection of spam, the spam content is isolated from legitimate reviews, and the spam content is then evaluated to determine the spam's nature. The system controller detects false spam, which the administrator will then remove. After the operation is completed, users can browse the final set of valid reviews and then purchase the products they choose from the website. [14-15]

Section1 of the paper states the Introduction, and also includes section2 consisting of Methods and Materials, section3 consists of Model and Analysis, section4 consists of Results and discussion, and section5 consists of Conclusion.

**2 Method and Materials**

***2.1 Methodologies***

The accompanying algorithms having taken awhile to implement:

A. Naïve Bayes Algorithm

Algorithms are used in software. Naïve in supervised machine learning, Bayes Classification is a technique. It's a simple classification approach, but it's also one of the most powerful. It is a theory based on assumptions. Even small violations of such premises have little effect on the algorithm's performance. Measuring the frequencies and combined values in a dataset, Naïve Bayes develops a collection of probabilities. Naïve Bayes is used for a variety of things, including text classification, spam filtering, and recommendation systems. In the given equation, A and B are instances. Probability of A occurring given evidence B has already occurred is denoted by P(A|B), P(A) is probability of A occurring, P(B) is probability of B occurring and P(B|A) is probability of B occurring given evidence A has already occurred.

(1)

***2.2 Data processing***

This section delves into the specifics of the suggested strategy depicted in Fig. 3.

***2.2.1. Pre-processing***

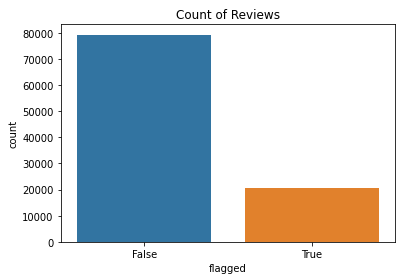
Data preparation is the initial stage in the suggested method, and it really is among the most important processes in machine learning methodologies. Data preparation is essential since real-world data is never appropriate for usage. To prepare the raw data from the clothing fabric and food review datasets for computational activity, a range of preprocessing approaches were used in this study. The following are the procedures:

1) Tokenization: The tokenization is among the extensively used Natural Language Processing methods. It's required before you can use any other preparation methods. Individual words that are segregated from the rest of the text are called tokens. Tokenization will break down a text like "Yaay! I have won a lottery " into tokens like: "Yaay", "I", "have", "won", "a", "lottery".

2) Stop Words Cleaning: The most commonly used meaningless keywords are called stop words. Stop words can be used in a number of different situations. Cleaning of stop words is done in this article before proceeding with the fake review identification technique.

3) Lemmatization: The lemmatization procedure can be used to transform a plural version of a singular version. Its goal mainly is excluding only the inflectional endings from the phrases and return to base or dictionary form. Changing the word "stays" to "stay".

Over one lakh reviews are shown in Fig. 1, with ‘False’ indicating authentic reviews and ‘True’ indicating false reviews.



**Figure 1:** Genuine and Fake reviews

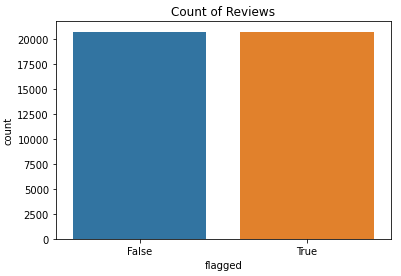
***2.2.2. Feature Extraction***

The purpose of feature extraction is to boost an object recognition or the performance of machine learning system. Feature extraction is the process of reducing data to its most important aspects, allowing computers and deep learning techniques to provide additional relevant information. It is primarily a strategy for reducing extraneous attributes from the data, which can reduce model accuracy.

Several approaches for extracting features for fake review identification have been developed in the literature. Using textual features is a common method. It includes sentimental analysis, which is based on the percentage of positively and negatively terms, such as "good" and "weak," in the review. The Cosine similarity is also taken into account. The similarity of cosine is the cosine angle formed by two N-dimensional vectors inside an N-dimensional environment divided by the product of their lengths. The true-false-inverse document frequency approach, which obtains the fraction of true and false answers as well as the inverse document frequency, is another textual feature method. Every phrase has a unique term frequency and inverse document frequency score, and a term's term frequency-inverse document frequency weight is the total of its term frequency and inverse document frequency scores. A matrix of confusion was used to categorize reviews into four parts: true negative: True events are labeled as such; true positive: False occurrences are labeled as such. Real actions are labeled as fake in false positive, while fake activities are labeled as real in false negative. The second set of features is the user's personal profile and behavioral characteristics. The two approaches to spotting scammers. Whether the comment of a user is more consistent and distinct than several other customers’ comments, or if the user simply repeats a review that has no link to other users. In two language models, term frequency-inverse document frequency is used to extract content features, primarily bi-gram and tri-gram.

***2.2.3. Feature Engineering***

Other descriptive elements of fake reviews are known to include the activities of the reviewers while composing their reviews. In this work, a few of these characteristics and how they affect the functionality of the false rating detection procedure are considered. Consider behavioral elements such as Maximum Content Similarity, punctuation, and plural numbers. Maximum Content Similarity, MNR, and RL, respectively, reflect the total number of capital letters used by a reviewer when composing the summary, the overall sum of punctuation marks discovered within every comment, and the number of total texts within every comment. And also, perform statistical analysis on reviewer behavior by using the "group-by" function, which calculates the quantity of bogus or actual reviews published by each reviewer on a specific day and for each hotel. When assessing the impact of users' behaviors on classifier performance, all of these characteristics are taken into account. The dataset has been under-sampled, as illustrated in Fig. 2.



**Figure 2:** Under-sampled data

***2.3 Modules***

Some of the modules used in the system to implement the models are as follows-

* NumPy: In Python, this library is used to do sophisticated mathematical operations.
* Pandas: It is a data analytics tool that lets users change data from large data sets.
* Matplotlib: It is used to display datasets and training results in terms of features and labels. It also enables us to visualize charts in order to better understand how algorithms learn.
* Sklearn: It is a Machine Learning package that includes techniques like Support Vector Machines (SVM), Random Forests, and K-Nearest Neighbors.
* NL TK: The Natural Language Toolkit Python library is a Python library that stands for Natural Language Toolkit. It is among the most widely used libraries for handling with language processing data, and it's also utilized in natural language processing.
* Tqdm: It is a package that makes it possible to be able to establish progress bars and estimate TTC instantly for given functions and loops.
* Seaborn: It is a library for data visualization that is based on Matplotlib. The model uses it to visualize the under-sampled data.

***2.4 Block Diagram***

The block diagram is shown in Fig. 3 below, that represents the proceedings in order to build our system. It gives an idea, starting from choosing datasets to eventually training models and evaluating them. Basically, the food-review dataset has been selected to start with. It has been used for tokenization, stop words cleaning, and lemmatization to preprocess the data. The feature engineering process is also used to build some features for making an easy evaluation. The data was shuffled to avoid biases, then was split into two sections. One for training and another for testing data.

Hyper Parameter tuning

Model Evaluation

**Dataset**

Stop Words Cleaning

**Testing**

**Learned Classifier**

Tokenization

**Training**

**- Random Forest**

Lemmatization

**Feature Engineering**

**Figure 3:** Block diagram

The suggested approach is assessed using the food review dataset. This dataset holds one lakh reviews of food by 70396 reviewers. The reviews are divided into two categories: legitimate reviews (79290) and bogus reviews (20704). The reviews were divided into two categories in the dataset: genuine and fraudulent. The Id, ProductId, UserId, ProfileName, HelpfulnessNumerator, HelpfulnessDenominator, Score, Time, and Summary are all included in each instance of the review in the dataset. The characteristics of the used dataset are summarized in Tab. 1 shown below.

The data has a maximum length of a review of 1913 words, a minimal length of a review of 5 words, an average length of a review of 51.2 words, a total number of tokens of 8308326 words, and the number of unique phrases of 56095 words.

**Table 1:** Summary of the dataset

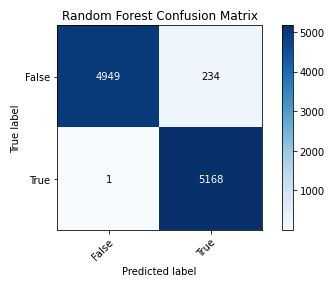
|  |  |
| --- | --- |
| Number of reviews in total | 99994 reviews |
| The total count of fraudulent reviews | 20704 reviews |
| The number of genuine comments | 79290 reviews |
| The number of unique phrases | 56095 words |
| Number of tokens in total | 8308326 words |
| The maximum length of a review | 1913 words |
| The minimal length of a review | 5 words |
| The average length of a review | 51.2 words |

**3 Model and Data Analysis**

The process approach is semi-supervised. The training dataset has been trained with machine learning algorithm - Random Forest, and tested the testing data with the same machine learning algorithm. The method that the software employs a supervised machine learning techniques called Random Forest classification. It is a straightforward way to classification, but it is also one of the most effective. It is a theorem based on a hypothesis. The algorithm's performance is unaffected by even minor violations of these assumptions. The applications of the algorithm include sentiment classification, spam detection, and recommendation engines, to name a few. Random Forest is a good solution for dealing with overfitting concerns in decision trees. The goal of a RF focuses to create a tree collection from various dataset samplings. RF manufactures a limited number of trees at random, rather than each tree in the forest being produced across all features.

After training and testing over the model, the performance has been evaluated and also compared to its accuracy. Finally, choose the machine learning classification algorithm that gives the best performance result from another. The model also retrieved other characteristics reflecting reviewer behaviors throughout the authoring of their reviews from the dataset and its statistics. These characteristics include 'RL,' which represents the total number of words used by a reviewer when writing a review, 'MNR,' that reflects the maximum count of reviews per day per reviewer, and 'Maximum Content Similarity,' which measures the number of plagiarism or copyright texts found. The system will take these variables into consideration in attempt to decide the implications of the users' answers on the models' performance.

The system shows a tabular summary of the number of true and false predictions made by a classifier using a confusion matrix. There are labels for false: false 5034, false: true 149, true: true 5168, and true: false 1 in Random Forest Confusion Matrix. On the other hand, in the Naïve Bayes Confusion Matrix, there are labels for false: false 4888, false: true 295, true: true 1381, and true: false 3788. For the Naïve Bayes algorithm, the true positive and f1-score is too low. On the other hand, Random Forest gives a better number of precision and recall. The f1-score also increased by 31.2%. Fig. 4 shows the confusion matrix of the Random Forest algorithm below.

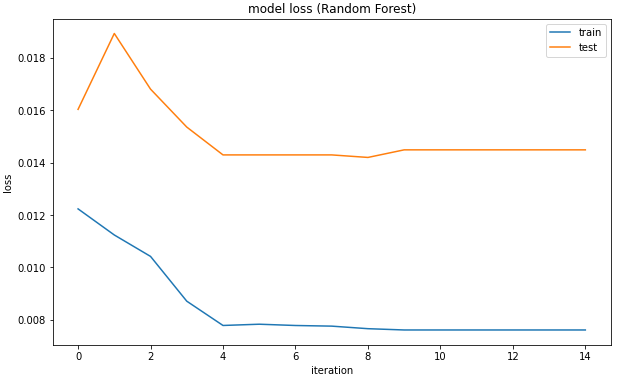


**Figure 4:** Random Forest Confusion Matrix

It depicts how True Positive and True Negative give a good score over five thousand data. Besides, the False positive have been decreased, which is not extreme for the unseen dataset.

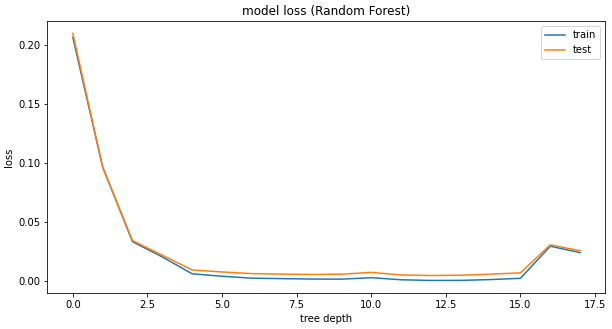
**4 Results and Explanation**

In the beginning the model was overfitted shown in Fig. 5 depicts the training loss and testing loss of Random Forest. It’s showing the decreasing of loss in training and testing using Random Forest algorithm. On the other hand, training error has decreased enough but testing error has not decreased rather it increased using Naïve Bayes algorithm. So that we only considered Random Forest algorithm to our model.



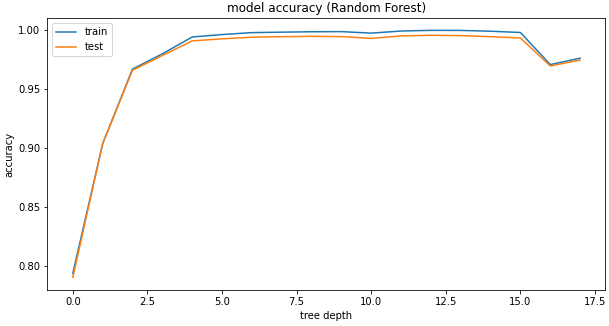
**Figure 5:** Before hyperparameter tuning model loss

It seems that the model has low bias and high variance in both algorithms shown in Fig 5 After hyperparameter tuning the overfitting problem is solved for both random forest and naïve bayes. The new training and testing loss of both algorithms shown below to Fig. 6 also the accuracy graph shown below Fig. 7.



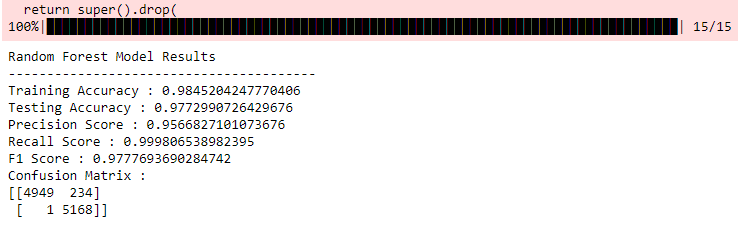
**Figure 6:** After hyperparameter tuning model loss

Model loss for both algorithms has improved after tuning the hyper parameters. In random forest, training accuracy becomes 0.97089 and testing accuracy becomes 0.96972. Similarly in naïve bayes, training accuracy becomes 0.78578 and testing accuracy becomes 0.72382 but it keeps changing and is not giving better performance. In conclusion, it can be said that the loss of training and testing of the model using the Random Forest algorithm is minimal and much better than the Naïve Bayes algorithm and we are not considering this algorithm.



**Figure 7:** After hyperparameter tuning model accuracy of Random Forest

The model was built using two different algorithms before, one of which is Nave Bayes and the other is The Random Forest. In the two following models below, the average f1-score is used to estimate the relative classification performance of both models. Random Forest surpasses the overall classifiers with an average f1-score of 97 %, according to the findings. Fig. 5 shows the precision of the result from training Random Forest algorithm we have not considered another algorithm.



**Figure 8:** Screenshot of the result

Tab. 2 shows recall, accuracy, f1-score, and therefore accuracy in the context of the users' retrieved traits and behaviors that is seen in Fig 6. From the above matrixes true and false labels are obtained and recall, precision and accuracy are calculated using the following equations-

(i)

(ii)

(iii)

The f1-score is used as the evaluation criterion of each classifier for the recall versus precision trade-off. The relative classification performance in both models is estimated using the average f1-score. According to the findings, Random Forest performs best among all other algorithms.

**Table 2:** Recall, Precision, F1-Score and Accuracy of the model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithms | Recall of algorithms | Precision of algorithms | F1-Score of algorithms | Accuracy of algorithms |
| Random Forest | 99.9% | 95.6% | 97.8% | **97.7%** |

In Tab. 2, Random Forest, with a f1-score of 97.8 %, it exceeds other predictors. The model has the following characteristics, the overall quality of the classifications is evaluated using the average f1-score. Random Forest surpasses the overall classifiers with an average f1-score, according to the findings. The outcome is increased by 31.2% when the features extracted are taken into consideration. It had the highest f-score. Finally, the findings show that Random Forest (with 15 iterations) surpasses the other classifier in terms of f-score, with the best f1-score. We have seen that training accuracy was 98.4 % and finally it came up with 97.7% model accuracy which is far better than the other model accuracy which used by different algorithms.

The Tab.3 compares all the algorithms mentioned in different papers with the machine learning approaches.

**Table 3:** Algorithm and Accuracy Comparison

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Number | Name | Dataset | Algorithm1 | Algorithm2 | Accuracy1 | Accuracy2 |
| 1. | This paper | Kaggle | Random Forest |  | **97.7%** |  |
| 2. | Ref [2] | Amazon | J48 Classifier |  | 93% |  |
| 3. | Ref [5] | Amazon | Support Vector Machine |  | 92% |  |
| 4. | Ref [7] | Amazon | AMT |  | 92% |  |
| 5. | Ref [9] | Yelp | J48 Classifier |  | 94% |  |
| 6. | Ref [21] | Amazon | Decision Tree | Naïve Bayes | 96.2% | 95.9% |
| 7. | Ref [22] | Yelp | Logistic regression | Random Forest | 86.9% | 86.8% |
| 8. | Ref [22] | Yelp | Support Vector Machine | KNN | 86.9% | 86.2% |

In 2, it provides a behavioral technique for detecting review spammers that attempt to manipulate product evaluations. For rank reviewers, an aggregated behavior rating technique has been developed, in order to detect online spam reviews. It has an accuracy of 93% and 3 recommends using categories of lexical, semantic, and linguistic aspects, having an accuracy of 92%. Using real-life filtered (fake) and not filtered (non-fake) Yelp reviews, 4 first made a comparison. The results revealed that real-world data is far more difficult to categorize, with an accuracy of only 67.8% later on.

In Tab. 3 no. 5 and 6, it is clear that sentiment analysis plays an important part in making product or service business decisions. Sentiment Analysis includes both text mining and information retrieval ideas. Feature weighting, which is critical for good classification, one of the really challenging components of sentiment analysis is determining the source of the sentiment. Soft computing approaches are also not widely used in literature, as may be observed. Life is like an empty vessel when you don't have an opinion. It gives 96.2% in the Decision Tree and 95.9% in Naïve Bayes algorithm. In 7 and 8, the results show that SVM, with a score of 86.9%, is the classifier with the greatest precision in Bi-gram. With scores of 86.9% and 86.8% respectively, Logistic Regression and Random Forest are rather close in accuracy. The models are put to the test first without the extracted features and then with obtained behaviors. In each instance, the accuracy of predictors in the scale of bigrams and trigrams classification algorithms is investigated.

**5 Conclusion**

The research demonstrated the significance of comments and that they have an impact on virtually every area of social media data. Reviews have a significant impact on people's opinions. As a result, detecting false reviews is a promising research topic. The research described a method for detecting phony reviews using machine learning, which recognizes both review features and reviewer behavior. To test the suggested technique, the food review dataset was employed. The approach is semi-supervised, with different classifiers being used. In the fraudulent review detection process, the results show that the Random Forest classifier outperforms another classifier. In addition, the results suggest that taking into account the reviewers' behavioral characteristics boosts the f-score by 55.5% and the ultimate accuracy to 97.7 %. In the current technique, not all reviewers' behavioral characteristics have been considered.

Other low-level features, such as frequent time or date dependency, the reviewer's timing for giving a review, and how common it is to deliver favorable or poor reviews, will be added in the future. More behavioral features must be held in order to improve the efficacy of the offered fake review detecting algorithm.

**Data Availability:** The data used to corroborate this study's findings is available for free at https://www.kaggle.com/laowingkin/amazon-fine-food-review-sentiment-analysis?select=Reviews.csv

**Acknowledgement:** The authors would like to express their gratitude to the Department of Electrical and Computer Engineering of North South University.

**Funding Statement:** “The research received no outside funding for this work.”

**Conflicts of Interest:** “The authors are glad to declare that they have no conflicts of interest to report regarding the present study.”

**References**

1. X. Yang, “One Methodology for Spam Review Detection Based on Review Coherence Metrics,” *International Conference on Intelligent Computing and Internet of Things (IC1T),* vol. 14, pp. 1-20, 2015.
2. D. D. Arifin, Shaufiah and M. A. Bijaksana, “Enhancing Spam Detection on Mobile Phone Short Message Service (SMS) Performance using FP-Growth and Naive Bayes Classifier,” *The 2016 IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob),* vol. 5, pp 144-150, 2016*.*
3. X. Fang and J. Zhan, “Sentiment analysis using product review data,” *Fang and Zhan Journal of Big Data (2015).*
4. Rastogi and M. Mehrotra, “Opinion spam Detection in Online Reviews,” *Journal of information and Knowledge Management*, vol. 16, no. 4, pp. 1-38, 2017.
5. J. Rout, S. Singh, S. Jena, and S. Bakshi, “Deceptive review detection using labeled and unlabeled data,” *Multimedia Tools and Applications*, vol.76, no. 3, pp. 3187-3211, 2016.
6. R. Narayan, J. Rout and S. Jena, “Review Spam Detection Using Semi Supervised Technique,” *Progress in Intelligent Computing Techniques: Theory, Practice, and Applications,* vol. 4, pp. 281-286, 2018.
7. M. I. Ahsan, T. Nahian, A. A. Kafi, M. I. Hossain, and F. M. Shah, “Review spam detection using active learning,” *2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON),* vol. 2, pp. 5-9,2016.
8. N. Jindal and B. Liu, “Review spam detection,” *Proceedings of the 16th international conference on World Wide Web*, ACM, vol. 2, pp. 1189–1190, 2007.
9. R. S. Jadhav and D. V. Gore, “New Approach for Identifying Manipulated Online Reviews using Decision Tree,” *(IJCSIT) International Journal of Computer Science and Information Technologies,* vol. 5, pp 1447-1450, 2014.
10. L. S. Chen and J. Y. Lin, “A study on Review Manipulation Classification using Decision Tree,” *IEEE conference publication*, vol. 8, pp 3-5, 2013.
11. N. Jindal, B. Liu, and E. P. Lim, “Finding unusual review patterns using unexpected rules,” in *Proceedings of the 19th ACM international conference on Information and knowledge management*, vol. 8, pp. 1549–1552, ACM, 2010.
12. T. Joachim, “Text categorization with support vector machines: Learning with many relevant features,” in *European conference on machine learning*, vol. 7, pp. 137–142, Springer, 1998.
13. E. P. Lim, V.A. Nguyen, N. Jindal, B. Liu, and H. W. Lauw, “Detecting product review spammers using rating behaviors,” in *Proceedings of the 19th ACM international conference on Information and knowledge management*, vol. 3, pp. 939–948, ACM, 2010.
14. N. Jindal and B. Liu, “Analyzing and detecting review spam,” *International Conference on Web Search and*

*Data Mining,* vol. 4, pp. 547-552, 2007.

1. N. Jindal and B. Liu, “Opinion spam and analysis,” *International Conference on Web Search and Data Mining*, vol. 9, pp. 219-230, 2008.
2. M. L. Zhang and Z. H. Zhou, “ML-KNN: A lazy learning approach to multi-label learning,” *Pattern recognition*, vol. 40, no. 7, pp. 2038–2048, 2007.
3. N. Suguna and K. Thanushkodi, “An improved k-nearest neighbor classification using genetic algorithm,” *International Journal of Computer Science Issues*, vol. 7, no. 2, pp. 18–21, 2010.
4. M. A. Friedl and C. E. Brodley, “Decision tree classification of land cover from remotely sensed data,” *Remote sensing of environment*, vol. 61, no. 3, pp. 399–409, 1997.
5. A. Punde, S. Ramteke, S. Shinde and S. Kolte, “Fake product review monitoring and removal and sentiment analysis of genuine reviews,” *International Journal of Engineering and Management Research Science Issues*, vol. 9, no. 2, 2019.
6. A. M. Elmogy, U. Tariq, A. Ibrahim and A. Mohammed, “Fake review detection using supervised machine learning,” *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 1, pp. 604-605, 2021.
7. G. G. Chowdhury, “Natural language processing,” *Annual review of information science and technology*, vol. 37, no. 1, pp. 51–89, 2003
8. J. J. Webster and C. Kit, “Tokenization as the initial phase in nlp,” in *Proceedings of the 14th conference on Computational* linguistics, vol 4, 1992, pp. 1106– 1110.
9. V. Singh, R. Piryani, A. Uddin, and P. Waila, “Sentiment analysis of movie reviews and blog posts,” in *Advance Computing Conference (IACC)*, vol. 2, pp. 893–898, 2013.
10. J. Ramos et al., “Using tf-idf to determine word relevance in document queries,” in *Proceedings of the first instructional conference on machine learning*, vol. 242, pp. 133–142, 2003.
11. C. Lee and D. A. Landgrebe, “Feature extraction based on decision boundaries,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 4, pp. 388–400, 1993.
12. R. Mihalcea, C. Corley, C. Strapparava *et al.*, “Corpus-based and knowledge-based measures of text semantic similarity,” in *AAAI*, vol. 6, pp. 775–780, 2006.
13. J. Ramos *et al*., “Using tf-idf to determine word relevance in document queries,” in *Proceedings of the first instructional conference on machine learning*, vol. 242, pp. 133–142, 2003.