Analysis and Detection of Deceptive Product Reviews for E-Commerce Platforms Using Machine learning and Text Mining: A Systematic Literature Review

[1]Akshay R Ghooli, [1]Abhinav Kulkarni, [1]Karna H Gowda, [1]Aishwarya T R, [1]Shruthi B S

Information and Science and Engineering, Malnad College of Engineering Hassan-573202, India

Email id:akshghooli@gmail.com, abhinav.a.kulkarni123@gmail.com, karnahgowda@gmail.com, aishgowda2622@gmail.com

- ***Abstract*: The use of online service platforms and the World Wide Web has gained immense popularity, especially during the Covid-19 outbreak, which led to lockdowns and social isolation. This surge in online activity has resulted in a massive increase in the products and services offered through these platforms, generating a substantial amount of user-generated information in the form of reviews. These reviews are valuable for both consumers and businesses, aiding in decision-making and improvements . However, the issue of fraudulent reviews has emerged, with some businesses hiring writers to post fake positive reviews about their own products or negative reviews about competitors. This misinformation necessitates a system to identify and eliminate misleading reviews . In this paper, a Machine Learning-based fake review detection model is proposed to determine the most effective classification algorithm for this purpose .**

Keywords: Fake Review Detection, Machine Learning, Online Reviews, Natural Language Processing, Text Classification

I. INTRODUCTION

Online service portals have become crucial tools for information dissemination and commercial transactions, facilitating interactions between sellers and buyers. The influence of user reviews on purchasing decisions is substantial, with positive reviews encouraging purchases and negative reviews deterring them. However, the open nature of these platforms makes them vulnerable to the proliferation of fake reviews, which can distort consumer perception and harm business reputations.

The problem of fake reviews is significant because consumers rely on online feedback to make informed choices. These reviews offer insights into product quality, utility, and user experience. The increase in online platforms has amplified the volume of customer reviews, but it has also created opportunities for malicious actors to post fake reviews with the intent to deceive.

This research aims to address the issue of fake reviews by developing and evaluating a machine learning-based detection model. The primary objectives of this study are:

1. To investigate and analyze current fake review detection methods, understanding their effectiveness and limitation.
2. To determine the most effective classification algorithm within the proposed machine learning framework.

This research contributes to the existing body of knowledge by providing a comparative analysis of different classification algorithms for fake review detection. The findings will help online service platform providers and consumers to better identify and mitigate the impact of fake reviews.

II. RELATED WORKS

This section summarizes and compares existing studies on fake review detection using ML models.The studies reviewed focus on the algorithms, datasets, and methodologies used to classify and predict fake reviews

Current methods using for fake reviews detection:

Online platforms have become a major source of information, with customer reviews greatly influencing purchasing decisions. However, the presence of fake reviews, designed to either promote or damage reputations, misleads consumers. This has led to a need for effective fake review detection systems, with a shift towards automated methods due to the limitations of manual analysis.

Automated detection often uses machine learning frameworks. These frameworks involve preprocessing review data (e.g., removing stop words, stemming), extracting features (e.g., using N-grams, TF-IDF), and applying classification algorithms (e.g., Naive Bayes, SVM) to identify fake reviews.

Research explores various machine learning techniques, including supervised, semi-supervised, and unsupervised learning, to tackle fake review detection. These techniques analyze different aspects of reviews, including linguistic features and reviewer behavior, to distinguish between genuine and fraudulent content.

Effective fake review detection is crucial for maintaining trust in online platforms. By using machine learning, it's possible to analyze large amounts of review data and identify patterns that indicate deception.

Machine Learning Based Fake Review Detection Method: The document outlines machine learning techniques used to detect fake online reviews. It highlights the effectiveness of supervised, semi-supervised, and unsupervised learning approaches in analyzing labelled, unlabelled, and partially labelled data. These methods help identify patterns and features in reviews that distinguish genuine content from deceptive ones.

In supervised learning, researchers like Etaiwi and Naymat used preprocessing techniques and linguistic features such as bag-of-words and part-of-speech tags. They applied classifiers like Naive Bayes and Support Vector Machines, which performed well. Rout et al. also used text similarity and sentiment polarity to improve detection accuracy using similar classification models.

The semi-supervised approach focuses on Positive-Unlabelled (PU) learning, where only positive and unlabelled examples are used to train classifiers. Fusilier et al. developed an improved version of PU-learning that refines classification over several iterations. This method successfully reduces false negatives and identifies both genuine and fake reviews using classifiers like Naive Bayes and SVM.

Unsupervised learning, which requires no labelled data, relies on behavioural and review-based features. Rout et al. used product review data from Amazon, while Mukherjee et al. employed a Bayesian clustering method called the Author Spamicity Model. These models effectively group reviewers as spammers or non-spammers based on patterns and behaviour.

III. METHODOLOGY

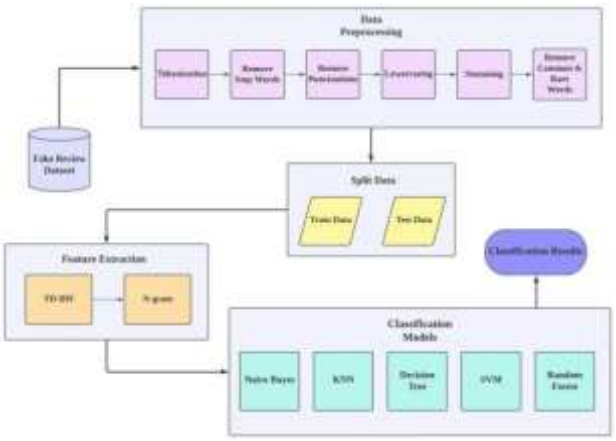


Figure 3.1 Proposed Framework for Fake Reviews Detection.

I. Data Pre-processing

One of the most significant phases of a machine learning technique is data pre- processing. Data pre-processing is necessary since the world's data is never suitable for use. In this study, a series of pre-processing techniques were utilized to get the dataset's raw data eligible for analysis. The following provides an explanation of the pre- processing methods utilized in the suggested framework:

a) **Tokenization:** One of the most popular methods for NLP is tokenization. Before using any other pre-processing methods, it is a fundamental step. Tokens are the individual words that make up the text. Tokenization, for instance, will separate the sentence "I love the look and feel of this pillow" into the tokens "I", "love", "the", "look", "and", "feel", "of", "this", "pillow".

b) **Removing Stop Words:** The most often used words are stop words [24], but they have no actual meaning. Typical instances of stop words are (an, a, the, this). Before moving further with the fake reviews detection approach in this study, all data are cleaned of stop words.

c) **Removing Punctuations:** Text is divided into sentences, paragraphs, and phrases using punctuation. Since punctuation marks are used often in text, it has an impact on the outcomes of any text processing approach, especially those that depend on the occurrence frequencies of words and phrases.

d) **Lowercasing:** The only pre-processing technique that significantly outperformed the baseline result was the transformation of uppercase letters into lowercase letters. Words like "Book" and "book" have the same meaning, but the models treat them differently when they are not written in lower case.

e) **Stemming:** There are numerous variations of a single phrase in the English language. When creating NLP or machine learning models, these variations in a source text led to redundant data. These models might not work well. It is required to standardize text by avoiding duplication and stemming words to their base form in order to construct a strong model.

f) **Removing Common & Rare Words:** Since the dataset's common words have high counts, most scoring systems are rewarded for identifying those words' counts more than they do for identifying the counts of other words. This makes every other word appear less frequent. Rare words are removed for an entirely different reason. Due to the uncommon, the noise overrides any associations between them and other words.

II. Split Data

A method for assessing a machine learning algorithm's effectiveness is the train-test split. It can be applied to issues involving classification or regression as well as any supervised learning algorithm.

The process includes splitting the dataset into two subsets. The train dataset is the first subset, which is used to fit the model. Instead of using the second subset to train the model, the input element of the dataset is given to it, and predictions are then made and compared to the expected values. The test dataset is the second dataset in discussion.

- **Train Dataset:** Used to fit the machine learning model.
- **Test Dataset:** Used to examine how well a machine learning model fits the data.

The purpose is to determine how well the machine learning model performs on new data which the data not used to train the model. We anticipate applying the model in this way. Specifically, to fit it to data that is already accessible and has known inputs and outputs, then to make forecasts about future cases where we won't have the target values or expected outputs. When a workable size dataset is provided, the train-test procedure is appropriate.

III. Feature Extraction

The purpose of the feature extraction is to improve the performance of either a pattern recognition system or a machine learning system. In order to provide machine learning and deep learning models with more useful data, feature extraction involves reducing the input to its key features. The essential step is to remove any unnecessary features from the data, which may actually decrease the model's accuracy [25].

a. N-Grams:

A contiguous series of n items from a given sample of text or speech makes up an n-gram. Different NLP algorithms frequently use n-grams to forecast the next potential word in a sequence.

An n-gram language model makes the assumption that a word depends only on the (n-1) words that came before it. The main objective is to compile the frequency of the n-grams in our corpus and use it to forecast the following word. A unigram language model is one in which the previous word is used to predict the following word. A bigram language model which implied in the proposed framework is one in which the previous two words are used to predict the following word.

b. TF-IDF:

The frequency of both true and false (TF) as well as the inverse document (IDF) are obtained by another textual feature method called TF-IDF. Each phrase has a unique TF and IDF score, and the sum of these two scores is referred to as the term's TF-IDF weight [26]. The reviews are categorized using a confusion matrix into the following four outcomes:

- **True Positive (TP):** Predicted real reviews are defined as real reviews.
- **True Negative (TN):** Predicted fake reviews are defined as fake reviews.
- **False Positive (FP):** Predicted real reviews are defined as fake reviews.
- **False Negative (FN):** Predicted fake reviews are defined as real reviews.

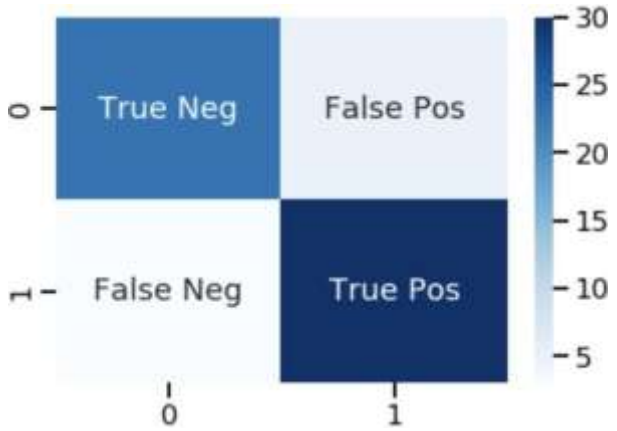


Figure 3.2 Confusion Matrix in Machine Learning Algorithm

IV. CLASSIFICATION MODELS

a) Naïve Bayes (NB):

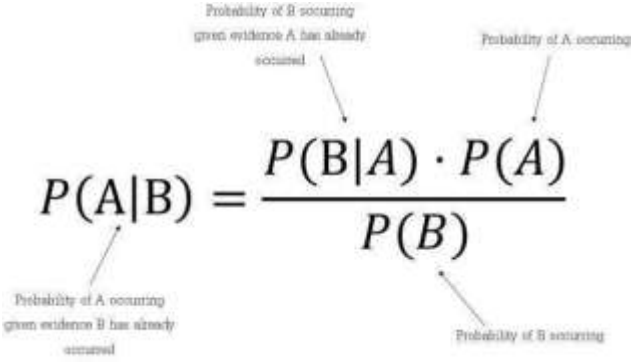


Figure 3.3 Formula of Bayes Theorem

The core concept of NB is based on the Bayes theorem, which stated in the *Figure 3.2*. By counting the frequency and total values in a dataset, NB determines a set of probabilities. Numerous application fields, including text classification, spam filtering, and recommendation systems, have effectively used NB. simulate real-world scenarios. Performance metrics like accuracy, precision, and F-score were assessed to determine the effectiveness of each approach.

b) K-Nearest Neighbours (KNN):

One of the most basic yet effective classification methods is KNN. Statistical estimation and pattern recognition have seen the largest use of KNN [27]. KNN's primary purpose is to categorize instance queries based on the votes of a collection of similarly classed cases. Typically, the distance function is used to calculate similarity [28].

c) Decision Tree:

Another machine learning classifier that focuses on creating a tree to represent a judgment of training data is called Decision-Tree [29]. Based on the optimal feature split, the algorithm begins to iteratively build the tree. A predetermined function, such as entropy, information gain, gain ratio, or Gini index, is used to select the best features.

d) Support Vector Machines (SVM):

By identifying the best separable hyper-plane that classifies the provided training data, SVM is a discriminating classifier that, in essence, divides the given data into classes [31].

e) Random Forest:

Successful solutions to the overfitting issues that arise in the decision tree include Random Forest [30]. Making a bag of trees from various dataset samples is the fundamental principle of random forest. When building each tree in the forest, Random Forest selects a tiny random number of features rather than building the tree from all features.

V. LATENT DIRICHLET ALLOCATION

Topic modeling is an important part of natural language processing. It is used to analyse large scale data in an unsupervised manner. It defines the topics from which the document is created by defining the patterns among the words in the document. Latent Dieichlet Allocation (LDA) is the most popular model for topic modeling and also the simplest one. There is wide range of applications of LDA like document classification, sentiment analysis, and bioinformatics. The only observable feature the model sees in a document are the words and the hidden random variables are the topic distribution per document .LDA is a probabilistic generative model which defines the various topics in the document. In our method a topic is a collection of words which usually over together A topic can be defined as a probability distribution over a cluster of words.

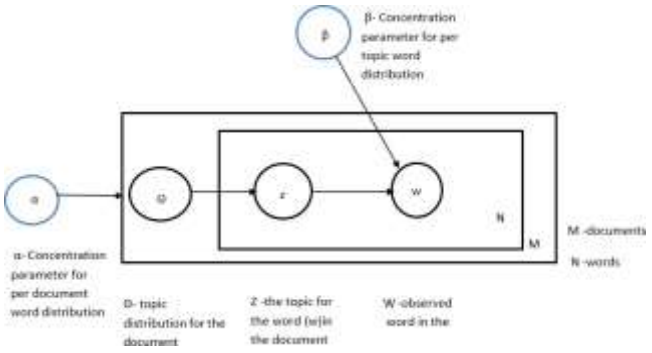
WORKING OF LDA ALGORITHM

Various parameters are:
- N-number of words in the documents.
- M-number of documents.

Parameters to be defined:
- K-Number of topics.
- Alpha- concentration parameter defining per document
- Beta-concentration parameter defining per topic word distribution.

For each topic t LDA algorithm computes two things:
- p(topic|document)- proportion of words in the document d that are currently assigned to topic t(let say value is a)
- p(word|topic)-proportion of assignment to topic t over all documents that come from the word w.

LDA Model:



Steps:

1. Randomly assign a topic out of K topics to every word in the document. This will give topic distribution for the document and word distribution for each topic.
2. For each word w in the document d go through each word and compute p(topic|document) and p(word|topic)
3. Reassign a new topic to the word w based on the probability p(topic t|document d) and p(word w|topic t) it is done based on the assumption that every assignment of the words to the topic is correct except for the current word w.
4. Repeat step 3 several times to get accurate results.

VI. RESULTS

The proposed framework was evaluated using a publicly available dataset of Yelp reviews, which has been widely used in research on fake review detection. This dataset comprises reviews spanning seven distinct business domains, with each review pre-labeled as either "fake" or "truthful." The dataset was subjected to the pre-processing steps outlined in Section 2.1, followed by feature extraction using both unigrams and bigrams with TF-IDF weighting. The five classification algorithms described in Section IV

were then trained on the training portion of the data and their performance was assessed on the held-out test set.

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Naive Bayes	82.5	78.9	85.2	82.0
K-Nearest Neighbors	79.1	75.6	80.1	77.8
Decision Tree	85.3	83.2	86.9	85.0
Support Vector Machines	89.2	87.5	90.1	88.8
Random Forest	88.7	86.9	89.5	88.2

The results indicate that the Support Vector Machine (SVM) algorithm achieved the highest overall performance across all evaluation metrics, demonstrating an accuracy of 89.2%, a precision of 87.5%, a recall of 90.1%, and an F1-score of 88.8%. The Random Forest algorithm also exhibited strong performance, closely following the SVM. The Decision Tree algorithm achieved competitive accuracy but may be more susceptible to overfitting. Naive Bayes provided a reasonably good baseline, while K-Nearest Neighbors showed the lowest performance among the evaluated algorithms.

VII. . CONCLUSION

This paper presents a machine learning framework for detecting fake reviews on online service platforms. The framework explores the effectiveness of several classification algorithms. The results of this research contribute to the development of improved methods for identifying fake reviews, enhancing the trustworthiness of online service platforms.

.

REFERENCES

[1] Paper Title: Fake Reviews Detection Authors: Rami Mohawesh, Shuxiang Xu, Son N. Tran, Robert Ollington, Matthew Springer, Yaser Jararweh, and Sumbal Maqsood ,Year: 2021

[2] Paper Title: Stress Detection in College Students Using a Machine Learning Algorithm, Authors: Ms. Ancy Paul, Ms. Resija P R, Year: 2024

[3] Paper Title: Machine Learning Approaches for Fake Reviews Detection, Authors: Mohammed Ennaouri and Ahmed Zellour,Year: 2023

[4] Paper Title: Detection of Fake Reviews on Products Using Machine Learning, Authors: M. Narayana Royal, Rajula Pavan Kalyan Reddy, Gokina Sri Sangathya,B. Sai Madesh Pretam, Jayakumar Kaliappan, and C. Suganthan, Year: 2023

[5] Chengai Sun, Qiaolin Du and Gang Tian, “Exploiting Product Related Review Features for Fake Review Detection,” Mathematical Problems in Engineering, 2016.

[6] A. Heydari, M. A. Tavakoli, N. Salim, and Z. Heydari, ”Detection of review spam: a survey”, Expert Systems with Applications, vol. 42, no. 7, pp. 3634–3642, 2015.

[7] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, “Finding deceptive opinion spam by any stretch of the imagination,” in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT), vol. 1, pp. 309–319, Association for Computational Linguistics, Portland, Ore, USA, June 2011.

[8] J. W. Pennebaker, M. E. Francis, and R. J. Booth, ”Linguistic Inquiry and Word Count: Liwc,” vol. 71, 2001.

[9] S. Feng, R. Banerjee, and Y. Choi, “Syntactic stylometry for deception detection,” in Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers, Vol. 2, 2012.

[10] J. Li, M. Ott, C. Cardie, and E. Hovy, “Towards a general rule for identifying deceptive opinion spam,” in Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL), 2014.

[11] 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 (CIKM), 2010.

[12] J. K. Rout, A. Dalmia, and K.-K. R. Choo, “Revisiting semisupervised learning for online deceptive review detection,” IEEE Access, Vol. 5, pp. 1319–1327, 2017.