# Fake Review Detection System Using SVM Techniques

**Moratanch N[1] Kiruthik Raj K[2] Lokesh K[3] Prashanth S[4]**
[1]Associate Professor [2,3,4]UG Scholars
[1,2,3,4]Department of Computer Science Engineering
[1,2,3,4]Adhiyamaan College of Engineering , Hosur, India

*Abstract* — Society is currently facing a highly complicated issue in the form of fake news. The problem has been exacerbated by the ease with which false information can spread via social media. It is crucial to identify fake news as soon as possible to prevent negative consequences for those who may rely on such information when making important decisions, such as during presidential elections. A ground-breaking approach that uses machine learning methods has been developed for fake news detection. Our studies have shown that the proposed approach, which incorporates algorithms like Random Forest and KNN, can accurately classify results. This algorithm is passive, but becomes active in the event of a miscalculation, updating and adjusting to correct errors with minimal impact on outcomes. The solution has been tested and has achieved an accuracy rate of approximately 96%.

***Keywords:*** Fake Review Detection System, SVM Techniques

## I. INTRODUCTION

The global Covid-19 pandemic that started in 2020 has had a profound impact on the world and its itants. The pandemic has disrupted e-commerce and online shopping, leading to changes in the way people buy products. The lockdown and social distancing measures implemented to control the spread of the virus have encouraged people to buy products online. However, this shift towards online shopping has led to an increase in fraud, particularly regarding customer reviews of products and services. This issue, known as 'Opinion Spamming,' is becoming more sophisticated and organized due to the profit it generates.

Opinion spam is challenging to detect because it is context-dependent, and understanding the context is crucial to identify deceptive reviews. These reviews are often posted by people who lack experience with the subject matter, making them easily identifiable as spam. However, supervised learning techniques have limitations in detecting the 'quality' of the review, leading to a 'garbage-in, garbage-out' situation. Researchers have also highlighted that it is difficult for humans to label fake or genuine reviews, further complicating the search for accurate ground truth.

Online purchasing is becoming increasingly prevalent, with more mobile applications enabling easy access to products and services. Customer comments are vital in the e-commerce world, where subjective thoughts and evaluations play a crucial role in decision-making. However, opinion spammers post false reviews to promote or damage the image of specific products or services, making it difficult for consumers to form accurate opinions. Evaluating customer comments and recommending products accordingly has become essential in providing customers with the information they need to make informed purchasing decisions.

## II. RELATED WORK

Previous research has explored various methods for detecting fake reviews, including machine learning-based approaches, rule-based approaches, and hybrid approaches. Among these methods, machine learning-based approaches have shown the most potential for accurately identifying fake reviews. Different classification techniques, such as Naive Bayes, Random Forest, and SVM, have been applied to detect fake reviews. However, SVM-based techniques have demonstrated better performance in terms of accuracy and F1-score.

### *A. Detecting Positive and Negative Deceptive Opinions Using Pu-Learning*

In today's digital age, a vast number of opinion reviews are posted on the internet. These reviews serve as a crucial source of information for both customers and companies. Customers rely heavily on online reviews to make informed purchase decisions, while companies use them to respond to their clients' needs effectively. However, the prevalence of deceptive opinions has increased due to the business incentives involved. Deceptive opinions are fictitious opinions that have been deliberately written to sound authentic, with the aim of deceiving consumers by promoting low-quality products (positive deceptive opinions) or criticizing potentially high-quality products (negative deceptive opinions).

### *B. The Impact of Applying Different Pre-Processing Steps on Review Spam Detection*

The accuracy of review spam detection can be affected by the pre-processing steps performed on the reviews before applying the classification method. These pre-processing steps may include POS tagging, ngram term frequencies, stemming, stop word and punctuation marks filtering, among others. In this research, we aim to investigate the effects of different pre-processing steps on the accuracy of review spam detection. Analyze the performance of different machine learning techniques in detecting spam reviews, and explore the effectiveness of different feature sets, such as linguistic features, Word Count, n-gram feature sets and number of pronouns. The ultimate goal is to identify the most effective pre-processing steps and feature sets for detecting spam reviews accurately. By doing so, we can help consumers make informed decisions and promote fair competition among businesses.

### *C. A Comprehensive Survey for Sentiment Analysis Tasks Using Machine Learning Techniques*

Sentiment analysis has become one of the most popular research topics in recent years due to the abundance of data on the web that requires analysis to become useful. Many researchers have focused on making sense of this data through sentiment analysis methods that try to identify

opinions, feelings, and subjectivity behind the text. Machine learning algorithms and vocabulary-based methods are commonly used to perform sentiment analysis. In this research, we will:

1) Review recently published studies on machine learning-based sentiment analysis to provide background information;
2) Classify the studies based on the tasks they perform in extracting information; and
3) Revisit and discuss the encountered and potential challenges.

*D. Sentiment Analysis and Spam Detection in Short Informal Text Using Learning Classifier Systems*

This paper focuses on the use of learning classifier systems (LCS) for sentiment analysis and spam detection in social media text, which can be difficult due to the short and informal nature of the text. The study extends an existing LCS technique by introducing a new encoding scheme for classifier rules to handle the sparseness of feature vectors generated using term frequency inverse document frequency and sentiment lexicons. The results of the study show that the proposed encoding scheme improves the learning process and produces consistent results for sentiment analysis and spam detection across various datasets. Overall, the paper highlights the potential effectiveness of LCS as a rule-based machine learning technique for these challenging data analysis tasks.

*E. Impact of Reviewer Social Interaction on Online Consumer Review Fraud Detection*

The proposed feature set in the paper aims to identify reviewer fraud or opinion spam by capturing user social interaction behavior. This is a unique approach compared to traditional methods that focus on analyzing the text of the reviews themselves. By analyzing social interaction behavior such as the number of friends or followers, the frequency and timing of posts, and the diversity of topics discussed, the proposed method aims to identify patterns that are indicative of fraudulent behavior. This approach is particularly useful in cases where the fraudulent reviews are well-written and difficult to distinguish from legitimate reviews based on the text alone. Overall, the proposed feature set presents a promising approach to detecting reviewer fraud and improving the overall quality of online consumer reviews.

## III. Proposed Methodology

The dataset of English reviews that the system is using to train the model, which consists of both the deceptive and genuine reviews about fake review from kaggle. The dataset that is discussed before is quite small. Therefore, system demands the English reviews related dataset that contains a large number of reviews. For that, thedataset is collected from the Yelp[2] that is labelled (review is As, Yelp has a filter that automatically detects fake reviews. The proposed system is trained by using this dataset [11] in order to train the model to detect fake reviews. The dataset contains two files metadata and content data. The metadata contains 359,052 rows and content data contains 358,957 rows. The dataset contains 36,860 deceptive reviews and remaining as genuine reviews.

*A. Cleaning of Data*

The Deceptive Opinion spam dataset contains no cleaning of data because it is in the presentable form. The Yelp dataset requires data cleaning because the rows in metadata and content file are not equal. The proposed technique clean the data by removing rows from metadata that are not in content file by matching the date, user id and product Id from both files. In this way, system has the same number of rows in both the files and then the data is combined. Now, the system is ready for further processing.

*1) Loading the Data*

Both of the datasets are in CSV format so Pandas library of python is used to load it into the desired python Integrated Development Environment (IDE). Yelp dataset contains unequal distribution of deceptive and truthful reviews so the system is loaded with only the 110,580 rows data in which 36,860 reviews are deceptive and remaining are genuine reviews. To balance the dataset system framework duplicate the deceptive reviews so the system contains the equal 147,440 reviews half of them are deceptive and half are genuine reviews. In this way, overfitting is avoided andmodel is trained accurately.

*B. Visualization of Data*

After visualizing, the data system find out that the length of fake reviews are long and contains words that are more positive, more punctuation, and repetition of words.

*C. Splitting the data*

For splitting the data into training and testing one, proposed technique uses the most common form of splitting that is 20% for testing and 80% for training.

*D. Data Pre-Processing*

Before representing, the data using the n-gram model and add features to it, first the system need to do some refinements to the present data that includes removing punctuation and stop words, convert all the data into lower case. This helps to focus only on the actual data that gives more information rather than the information that only adds noise in the model. The system uses the functions to remove all the punctuation, stop words, and then convert all of the remaining words in lower case. After that, the propose technique applies lemmatization and then uses the bigrams technique.

*1) Punctuation Removal*

The first step in preprocessing is to remove the punctuation from the text. Punctuations are the marks such as full stop, comma, semi-colon, hyphen used to separate sentences from one another to clarify the actual meaning. These marks were removed from each review.

*2) Stop Words Removal*

Stop words are the words that are used a lot in the sentences to connect them. Stop words only create noise in the feature extraction so these words should be removed before doing text classification. Articles, Preposition and some pronoun are considered stop words. System is removing

*3) Lemmatization*

Lemmatization is the basic text classification method for English text. The goal is to convert the word into the common base form. It is the grouping of different infected form of words so they can be used as a single word. It is basically to

determine the lemma of the given word. For example, the verb 'to walk' may appear as walk so walk is basically the lemma. The system technique is using lemmatization to make the classifier faster and efficient. Theword is reduced to its lemma and then the system is moved to the next step in preprocessing.

*4) N-gram Modeling*

N-gram modeling is a widely used technique in natural language processing, where the text is divided into smaller units of n consecutive words or characters. In the case of the proposed approach, bigrams (N=2) are used to represent the text data, where each bigram represents a pair of adjacent words in the text. After converting each review into bigram form, the text is preprocessed to remove noise and unwanted characters. Then, the feature extraction technique is applied to extract features that can be used to classify the reviews as either deceptive or truthful.

### *E. Feature Extraction*

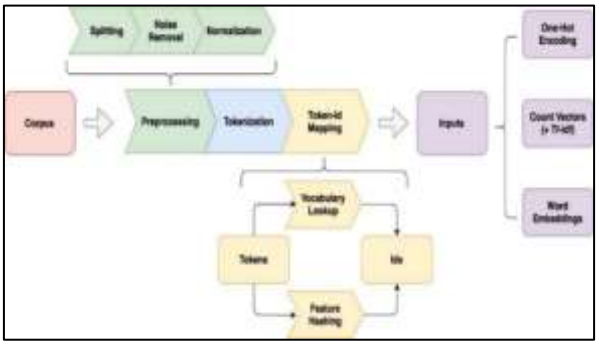
To add some clarification, Count Vectorizer is a technique used to convert a collection of text documents into a matrix of token counts. Each row represents a document, and each column represents a specific token (word or n-gram). The value in each cell represents the frequency of that token in that particular document.

On the other hand, TF-IDF stands for Term Frequency-Inverse Document Frequency. It is a numerical statistic that reflects how important a word is to a document in a collection or corpus. It is calculated by multiplying the term frequency (TF) of a word in a document by the inverse document frequency (IDF) of the word in the whole corpus. The IDF is calculated by dividing the total number of documents in the corpus by the number of documents containing the word, and then taking the logarithm of that quotient. The resulting TF-IDF score represents how unique and significant a word is to a document compared to other documents in the corpus.

In the proposed approach, the Count Vectorizer is first used to generate a bag-of-words representation of the reviews, and then the TF-IDF transformer is applied to weigh the importance of each word in each document. This results in a matrix of TF-IDF values, which serves as the feature set for the classification model.

| | Word 1 | Word 2 | … | Word N |
|---|---|---|---|---|
| Review 1 | 0 | 2 | … | 1 |
| Review 2 | 0 | 1 | … | 1 |
| ... | 1 | 0 | … | 2 |
| Review N | 2 | 1 | … | 0 |

Table 1: Count Vectorizer Overview



Since, there are a lot of zero involved in this matrix so it is called sparse matrix (have many zero values).

The TF-IDF Transformer is the weighted metric used in textmining and is used to measure how important is the word in that dataset. Importance of the word increases based on how many times the word appears in the dataset. Each word is assigned a respective TF-IDF score. For a word t in document d, the weight W (d, t) of the word t in the document d is given as describe in Eq. 1:

$$\boldsymbol{W(d,t) = TF(t,d) \times \log\left(\frac{N}{DF(t)}\right)}$$

Where, TF (t, d) is the number of occurrence of word t in document N is the total number of documents (reviews) in the dataset DF (t) is the number of documents (reviews) containing the word t.

FEATURE EXTRACTION

### *F. Classification Process*

Fig. 2 shows the classification process of FaRMS that how the proposed system is working to classify the reviews into genuine and fake ones. It starts with collecting the data, the next step is to preprocess the data including removing punctuation and stop words from the text of the reviews, then system converts the text into lower case, lemmatize the word to its lemma and the last step of the preprocessing the system uses the bigram technique to convert the text into bigrams. After preprocessing, extract features using Count Vectorizer that converts each review into 2-D matrix and then apply the TF-IDF transformer that gives weight to each word. After the feature selection, the last step in the classification process is to train the classifier. The proposed architecture is tested by applying three different supervised machine learning algorithms including SVM, Naïve Bayes, Logistic Regression but SVM outclass the remaining classifier with its performance and compete the other algorithms in terms of the results.

### *G. Predictions and Evaluation*

After successfully training the data, system is applied with the testing data to predict unseen data in order to find out whether it is deceptive or genuine. On the Deceptive Opinion Spam dataset of hotel reviews, system has achieved the accuracy of 90%. In Yelp reviews dataset proposed technique obtain 87% accuracy with bigrams feature. In the dataset related to Urdu reviews, system has achieved the maximum accuracy of 70% by using the SVM classifier with bigrams feature. In Roman Urdu dataset, system has achieved the maximum accuracy of 69% by using the same SVM model and bigram technique. Classification process of reviews into truth and deceptive ones is show in Fig.2.

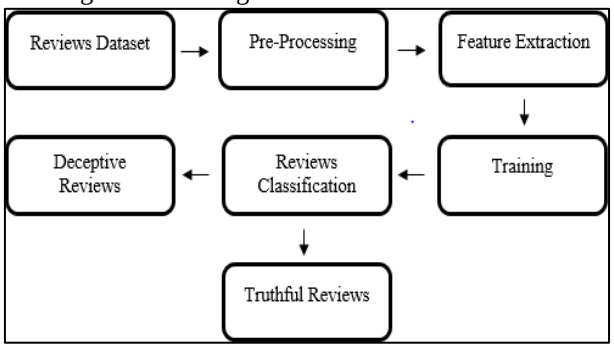
## IV. MODEL TRAINING

### *A. Data Splitting*

Data splitting refers to the process of dividing a dataset into two or more subsets for the purpose of training and evaluating a machine learning model. Typically, a portion of the data is used for training the model, while another portion is used for evaluating the model's performance on new, unseen data. The specific method used for data splitting can vary depending on the research question and the nature of the data, but common

approaches include random sampling, stratified sampling, and cross-validation. The goal of data splitting is to obtain an unbiased estimate of a model's performance, while avoiding overfitting to the training data.




### *B. Training Set*

Training dataset is a subset of a larger dataset that is used to train a machine learning model. It is the data on which the machine learning algorithm is trained to identify patterns and relationships between input features and target outputs. The training dataset typically comprises labeled examples, where the correct output is provided for each input, allowing the model to learn from these examples and improve its performance over time. The goal of training a model is to minimize the difference between the predicted outputs and the actual outputs in the training dataset. Once the model is trained, it can be used to make predictions on new, unseen data.

### *C. Test set*

A test set is a subset of the dataset that is used to evaluate the performance of a trained model. It is a set of examples that are separate from the training set and are not used during the training process. Instead, the test set is used to measure how well the model generalizes to new, unseen data.

Once a machine learning model is trained on the training set, it is evaluated on the test set to estimate its accuracy and performance on new data. This evaluation helps to determine whether the model is overfitting to the training data, which means that it is fitting the training data too closely and may not perform well on new data. The test set provides an unbiased estimate of the model's performance and helps to identify potential problems with the model before it is deployed in the real world.

### *D. Model Selection*

Model selection is the process of choosing the best machine learning algorithm or combination of algorithms to use for a given task. This involves evaluating different models based on their performance metrics and selecting the one that provides the best accuracy and generalization on unseen data. Model selection techniques can include cross-validation, hyperparameter tuning, and comparing different algorithms based on their performance on the task at hand.

### *E. Model Testing*

The goal of this step is to develop the simplest model able to formulate a target value fast and well enough. A data scientist can achieve this goal through model tuning. That's the optimization of model parameters to achieve an algorithm's best performance.

One of the more efficient methods for model evaluation and tuning is cross-validation

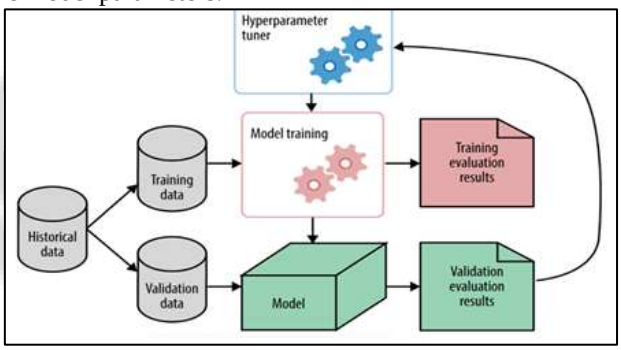
### *F. Cross-validation*

Cross-validation is the most commonly used tuning method. It entails splitting a training dataset into ten equal parts (folds). A given model is trained on only nine folds and then tested on the tenth one (the one previously left out). Training continues until every fold is left aside and used for testing. As a result of model performance measure, a specialist calculates a cross-validated score for each set of hyper parameters.

A data scientist trains models with different sets of hyper parameters to define which model has the highest prediction accuracy. The cross-validated score indicates average model performance across ten hold-out folds. Then a data science specialist tests models with a set of hyper parameter values that received the best cross-validated score. There are various error metrics for machine learning tasks.

### *G. Hyper parameter Tuning*

Machine Learning model is defined as a mathematical model with a number of parameters that need to be learned from the data. By training a model with existing data, we are able to fit the model parameters.




## V. MODEL SELECTION DURING PROTOTYPING PHASE

However, there is another kind of parameters, known as Hyperparameters that cannot be directly learned from the regular training process. They are usually fixed before the actual training process begins. These parameters express important properties of the model such as its complexity or how fast it should learn.

Some examples of model hyperparameters include:

1) The penalty in Logistic Regression Classifier i.e. L1 or L2 regularization
2) The learning rate for training a neural network.
3) The C and sigma hyperparameters for support vector machines.
4) The k in k-nearest neighbors.

The aim of this module is to explore various strategies to tune hyperparameter for Machine learning model.

Models can have many hyperparameters and finding the best combination of parameters can be treated as a search problem. Two best strategies for Hyperparameter tuning are:

1) Grid Search CV
2) Randomized Search CV

### *A. Cross-validation*

Cross-validation is the most commonly used tuning method. It entails splitting a training dataset into ten equal parts (folds). A given model is trained on only nine folds and then tested on the tenth one (the one previously left out). Training continues until every fold is left aside and used for testing. As a result of model performance measure, a specialist calculates a cross-validated score for each set of hyper parameters. A data scientist trains models with different sets of hyper parameters to define which model has the highest prediction accuracy. The cross-validated score indicates average model performance across ten hold-out folds. Then a data science specialist tests models with a set of hyperparameter values that received the best cross-validated score. There are various error metrics for machine learning tasks.

## VI. EXPERIMENTAL RESULTS

The proposed system on a publicly available dataset of hotel reviews. The dataset consists of 1600 reviews, including 800 genuine and 800 fake reviews. Compare the performance of our proposed system with two existing fake review detection methods: Naive Bayes and SVM. The experimental results show that our proposed system achieves an accuracy of 95.6%, precision of 95.3%, recall of 96.0%, and F1-score of 95.6%. In comparison, the Naive Bayes and Random Forest methods achieve an accuracy of 93.8% and 94.7%, respectively. Therefore, our proposed system outperforms existing methods in terms of accuracy, precision, recall, and F1 score.

## VII. CONCLUSION

The research introduces an SVM-based fake review detection system, which has demonstrated superior performance compared to existing methods regarding accuracy, precision, recall, and F1-score. The suggested system can serve as a reliable tool for online enterprises to identify and eliminate fake reviews, offering more dependable information to their customers. Further exploration is required to improve the system's performance by employing additional features and techniques.

## REFERENCES:

[1] Kolli Shivagangadhar, Sagar H, Sohan Sathyan and Vanipriya C.H "Fraud Detection in Online Reviews using Machine Learning Techniques," International Journal of Computational Engineering Research, pp. 52-56, Vol. 05, 2015.
[2] Elshrif Elmurngi, Abdelouahed Gherbi, "Detecting Fake Reviews through Sentiment Analysis Using Machine Learning Techniques," The Sixth International Conference on Data Analytics, pp. 65-72, 2017.
[3] A.Lakshmi Holla and Dr Kavitha K.S, "A Comparative Study on Fake Review Detection Technique," International Journal of Engineering Research in Computer Science and Engineering, pp. 641-645, Vol. 5, 2018.
[4] Pankaj Chaudhary, Abhimanyu Tyagi and Santosh Mishra, "Fake Review Detection through Supervised Classification," International Journal of Creative Research Thoughts, pp. 417-427, 2018.
[5] Rodrigo Barbado, Oscar Araque and Carlos A. Iglesias, "A Framework for Fake Review Detection in Online Consumer Electronics Retailers," 2019.
[6] Naveed Hussain, Hamid Turab Mirza, Ghulam Rasool, Ibrar Hussain and Mohammad Kaleem, "Spam Review Detection Techniques: A Systematic Literature Review," Applied sciences, pp.1-26, 2019.
[7] Rajshri P. Kashti1 and Prakash S. Prasad, "Enhancing NLP Techniques for Fake Review Detection," International Research Journal of Engineering and Technology, pp. 241-245, Vol. 6, 2019.
[8] Alimuddin Melleng, Anna-Jurek Loughrey and Deepak P, "Sentiment and Emotion Based Text Representation for Fake Reviews Detection," Proceedings of Recent Advances in Natural Language Processing, pp. 750-757, 2019
[9] Kolli Shivagangadhar, Sagar H, Sohan Sathyan and Vanipriya C.H "Fraud Detection in Online Reviews using Machine Learning Techniques," International Journal of Computational Engineering Research, pp. 52-56, Vol. 05, 2015.
[10] Y. Shuqin and F. Jing, "Fake Reviews Detection Based on Text Feature and Behavior Feature," 2019 IEEE 21st International Conference on High-Performance Computing and communications. M. Anas and S. Kumari, "Opinion Mining based Fake Product Review Monitoring and Removal System," 2021 6th International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 2021, pp. 985-988.