ISSN: 2229-6956 (ONLINE)
DOI: 10.21917/ijsc.2023.0444

DETECTING DECEPTIVE REVIEWS: AN INTEGRATED MACHINE LEARNING APPROACH

Anusuya Krishnan[1] and Kennedyraj[2]
[1]*College of Information Technology, United Arab Emirates University, United Arab Emirates*
[2]*Department of Information Technology, Noorul Islam University, India*

Abstract

In recent years, online reviews have become a crucial factor in promoting products and services. However, the rise of fake reviews has posed a significant challenge. Businesses, marketers, and advertisers often resort to embedding fake reviews to attract customers or undermine their competitors. Deceptive reviews have become a common practice, as they serve as a means of promoting one's own business or tarnishing the reputation of rivals. Consequently, the identification of deceptive reviews has emerged as a critical and ongoing research area. This research paper presents a machine learning model approach to detect deceptive reviews. The study focuses on experiments conducted using a deceptive opinion spam corpus dataset, specifically targeting restaurant reviews. An n-gram model combined with max features is developed to identify deceptive content, with a particular emphasis on fake reviews. Additionally, a benchmark study is conducted to explore the performance of two different feature extraction techniques and their application in five machine learning classification techniques. The experimental findings demonstrate that the passive aggressive classifier outperforms other algorithms, achieving the highest accuracy not only in text classification but also in identifying fake reviews. Moreover, the research delves into the identification of deceptive reviews and explores diverse feature extraction and machine learning techniques to improve the model's accuracy.

Keywords:

Natural Language Processing, Transformers, Deceptive Reviews

1. INTRODUCTION

With the continuous expansion of e-commerce platforms, the exchange of opinions and the abundance of online reviews related to products have seen a significant rise. Platforms like Amazon, Volusion, Shopify, BigCommerce, Magento, WooCommerce, Wix, and Big Cartel allow individuals to share their opinions about specific products, providing valuable insights for potential buyers. When contemplating online purchases, customers often rely on these reviews to gain a deeper understanding of the products they intend to buy. By analyzing the opinions of other customers on e-commerce websites, new customers can make informed decisions about whether to proceed with a purchase. Positive feedback from online reviews often encourages customers to buy a product, highlighting the importance of these reviews as a source of information. However, alongside genuine reviews, deceptive practices have emerged, where individuals intentionally post misleading reviews to promote or undermine the reputation of particular products. These deceptive reviews, also known as fake reviews, can distort customer perceptions and manipulate their purchasing decisions [1].

Businesses may engage in this practice by providing inauthentic content to sway customers' opinions. Individuals who engage in posting misleading opinions are often referred to as opinion spammers. However, amidst this growth, the prevalence of fake reviews has outpaced the improvement in the overall quality of online reviews. The escalation of malicious false reviews has led to an increasing number of instances where both retailers and customers suffer harm. As a consequence, users are finding it increasingly challenging to discern helpful reviews amidst the overwhelming flood of information. This blurring of the intrinsic value of online reviews, which traditionally assists in reducing uncertainty during pre-purchase decisions, has resulted in a decline in the credibility and traffic of e-commerce platforms [1].

Online reviews serve as a vital source of information for customers seeking insights into products they intend to purchase. Customers share their experiences through reviews, both positive and negative, influencing businesses in the long run. Unfortunately, this environment creates opportunities for the manipulation of customer decisions through the generation of false or fake reviews, known as opinion spamming. Spammers deliberately write deceptive opinions to sway others. These reviews can either aim to enhance or damage the reputation of a business or product [2].

Deceptive reviews can be broadly categorized into three groups. Firstly, there are untruthful reviews that purposefully provide false information about a product to either boost or tarnish its reputation. The second group includes reviews that target the brand without expressing any experience with a specific product. The third group consists of non-reviews and advertisements that contain text indirectly related to the product. Identifying groups two and three is relatively straightforward, while the detection of group one is more challenging. Such reviews may be authored by an individual spammer hired by a business owner or a collective effort of spammers working together within a specific timeframe to manipulate the reputation of a product or store [3]-[4].

Moreover, it has become evident that opinion spamming extends beyond product reviews and customer feedback. This paper sheds light on the significance of online reviews as crucial sources of information for customers within the e-commerce industry. It also highlights the challenges posed by deceptive reviews, which seek to misdirect and manipulate consumers. Understanding the impact of fake reviews and the presence of opinion spammers is essential for maintaining the integrity and credibility of online reviews, ensuring that customers can make well-informed purchasing decisions [4].

The subsequent sections of this paper provide a comprehensive overview of the research findings. Section 2 offers a summary of related works in the field, providing insights from existing literature. In Section 3, we delve into the background and intricacies of our proposed machine learning approach for detecting deceptive reviews. Section 4 presents the details of two experiments conducted to assess the accuracy of our model in

identifying deceptive reviews. The outcomes of these experiments are discussed, providing valuable insights into the effectiveness of our approach. Finally, Section 5 concludes the paper by summarizing the key findings and contributions of our work, while also outlining potential areas for future research and development.

2. RELATED WORKS

In this segment, we see some existing works related to opinion spam detection and the various different methods used by researchers to detect fake reviews. The opinion spam data was the first fake review dataset in Amazon reviews and provided some benchmark solution to detect them [5]-[7]. There are two main categories of opinion spam detection research approach: review spam (textual) and review spammer (behavioural). The author conducted a study focusing on textual-based detection of deceptive reviews. They utilized n-gram analysis and term frequency as feature extraction techniques to identify deceptive reviews in a dataset collected from Amazon Mechanical Turk and TripAdvisor. The dataset consisted of deceptive reviews of Chicago hotels and honest reviews, categorized into positive and negative groups. By implementing these techniques, they achieved an accuracy of 86% using a support vector machine (SVM) classifier [1]-[4].

In a separate study, researchers built upon opinion spam dataset work and utilized their dataset for detecting fake reviews [8]. However, they noted that machine-generated reviews do not accurately represent real-world deceptive reviews, as they do not reflect opinion spam. To validate their model, they tested it on a Yelp dataset and also employed opinion spam model for comparison. The results showed that the model trained using machine-generated fake reviews achieved only 67.8% accuracy, indicating its limitations in detecting real-world deceptive reviews. However, they emphasized that n-gram features still hold value in identifying deceptive reviews. The authors introduced a novel technique called the burst detection mechanism to identify customers who provide deceptive reviews [9]. They utilized a Markov random field model and incorporated a loopy belief propagation method to detect deceptive spammers within candidate bursts. The proposed approach achieved an accuracy of 77.6%.

In another related approach, researchers focused on detecting singleton reviews (SR). Their study revealed that over 90% of reviewers only post one review. Singleton reviews tend to have a significantly larger size compared to non-singleton reviews. The authors also observed that a rapid increase in the number of singleton reviews within a short duration indicates potential manipulation by fake reviewers aiming to influence a product's reputation or rating.

Another study concentrated on reviews that received a high number of reviewer's votes and comments. They hypothesized that reviews with a low number of votes are more suspicious and likely to be fake. The authors investigated supervised machine learning techniques, such as SVM, NB, and LR, to identify review spam. Using the NB method, they achieved an F-score of 0.58, which outperformed other methods relying on behavioral features [10].

Some studies discovered a correlation between distribution anomaly and fake review detection. They posited that certain business entities may hire spammers to write fake reviews [11]. They evaluated their approach using opinion spam's "gold-standard dataset," consisting of 400 deceptive and truthful reviews. They achieved an accuracy of 72.5% on their test dataset, highlighting the effectiveness of detecting suspicious bursts within a specific time window. However, their method is less effective in determining the authenticity of user reviews [12].

One of the studies developed an author spamicity model (ASM) to identify suspicious spammers based on their behavioral patterns. They categorized reviewers into spammers and non-spammers and proposed an unsupervised Bayesian inference framework for detecting deceptive reviews. Their results demonstrated that the ASM model outperformed other supervised machine learning approaches, highlighting its efficiency in detecting deceptive reviews [13]. The study suggested the potential application of K-Nearest Neighbors (KNN) algorithms to enhance the accuracy of the model. However, there are limitations to the versatility of the emotional dictionary, as it is specialized in the movie domain. Therefore, improvements are required to address this limitation. Additionally, the study had a technical limitation in that it did not incorporate adverbs, which play a crucial role in representing the intensity of emotional expression words [14].

The extraction of opinion data involves analyzing text data that expresses the user's opinion about the object itself or specific features of the object within a sentence, such as "This software updates really fast." In the case of Korean, this method utilizes preprocessing steps such as "morphology analysis" and "phrase analysis." Through proper natural language processing, including morpheme analysis and parsing, the opinion text extracts information such as the expression of the opinion, the object being referred to, and any modifiers. It then determines whether specific words have a positive or negative meaning and adjusts the strength of the meaning to derive a final polarity value [15].

OLSAP (Opinion-oriented Linear Sentiment Analysis Platform) views the polarity information of opinion data as a factor comparable to a "measurement," such as sales volume. By utilizing the polarity information in the opinion data, OLSAP captures complex information, including overall user evaluations of products, regional assessments, and temporal changes in opinions [16].

Overall, these studies present various approaches to detect deceptive reviews, encompassing burst detection mechanisms, analysis of singleton reviews, consideration of reviewer's votes, examination of distribution anomalies, and author spamicity modeling. Each approach brings valuable insights and contributes to the ongoing research in the field of deceptive review detection.

3. METHODOLOGY

The typical procedure for detecting fake reviews begins with preprocessing the dataset, which involves removing unnecessary special characters, punctuation, stop words, and irrelevant words. Following that, lemmatization is applied to extract features from the cleaned dataset. The final step in the classification process is training the classifier using the extracted features. In our study, we evaluated five distinct machine learning algorithms: support

vector machine (SVM), linear support vector machines (LSVM), passive aggressive classifier (PA), logistic regression (LR), and multinomial naive Bayes (NB).

3.1 DATA PREPROCESSING

Data cleaning or pre-processing is a crucial step in any machine learning task, especially when dealing with unstructured data. This process involves various techniques, such as removing punctuation, URLs, stop words, lowercasing, tokenization, stemming, and lemmatization. By applying these techniques to remove irrelevant information, we prepare the data for feature extraction.

3.1.1 Tokenization:

Tokenization is a fundamental technique used in natural language processing. It precedes other language processing methods and involves dividing the given text data into smaller units called tokens. Tokens can be alphanumeric characters, punctuation marks, or other special characters. In the case of a sentence like "the food is tasty," tokenization would produce the tokens: "the," "food," "is," "tasty."

3.1.2 Stop Words Removal"

Stop words are commonly used words in everyday English language, such as articles, conjunctions, interjections, prepositions, and some pronouns. Common examples of the stop words are for, from, how, in, is, of, on, or, that, the, these, this, too, was, what, when, where, who, will, and so on. These words don't carry significant meaning and are removed from each text document during pre-processing.

3.1.3 Lemmatization:

Lemmatization is the process of converting tokenized words into their base or root forms, making them more understandable to humans. It reduces words to their common root form, eliminating inflectional variations. For example, words like "singing," "sang," and "singer" would be reduced to the word "sing." Although lemmatization can be time-consuming, it is highly effective and commonly used in chatbot applications. In this paper, we employ lemmatization for data preprocessing.

3.2 FEATURE EXTRACTION

To input text data into our machine learning model, we need to convert words into numerical or vector form. Hence, it is essential to perform feature extraction to reduce the dimensionality of the text features. In this study, we utilized two feature extraction methods: count vectorizer (bag of words) and term frequency-inverse document frequency (TF-IDF). Below, we provide a brief explanation of each method.

3.2.1 Count Vectorizer:

Count vectorizer allows us to transform variable-length texts into fixed-length vectors and n-grams. It uses the bag-of-words (BoW) technique to represent a text as a vector of numbers. In this approach, a text is represented by a matrix, where each word corresponds to a column and each row corresponds to a sample text from the document.

3.2.2 Term Frequency-Inverse Document Frequency (TF-IDF):

While the bag of words method is simple and effective, it treats all words equally, without considering their importance. To address this limitation, we employ TF-IDF. TF-IDF is a feature extraction technique widely used in natural language processing. It measures the significance of a word within a document relative to the entire corpus. TF-IDF transforms word into vector form by multiplying the term frequency (TF) with the inverse document frequency (IDF). Term frequency (TF) is calculated by dividing the number of occurrences of a word in a document by the total number of words in that document. It can be expressed in Eq.(1):

$$TF = N/T \tag{1}$$

where N is the number of repeated words in the corpus and T is the total words present in the corpus. Inverse Document Frequency (IDF) is determined by taking the logarithm of the ratio between the total number of documents in the corpus and the number of documents in the corpus containing the specific word. It can be written as follows in Eq.(2):

$$IDF = \log T/N \tag{2}$$

where N is the number of repetitive words in the corpus and T is the total number of words present in the corpus. TF-IDF is obtained by multiplying TF with IDF in Eq.(3):

$$\text{TFIDF} = TF \times IDF \tag{3}$$

By employing TF-IDF, we address the limitations of the bag-of-words approach and capture the importance of words in the document relative to the entire corpus.

3.3 IMPLEMENTATION

Once the features have been extracted using either count vectorizer or TF-IDF, we proceed to train a machine learning model to classify reviews as truthful or fake. Prior to feature extraction, we partition the dataset into training and test sets using train-test split and K-Fold cross validation.

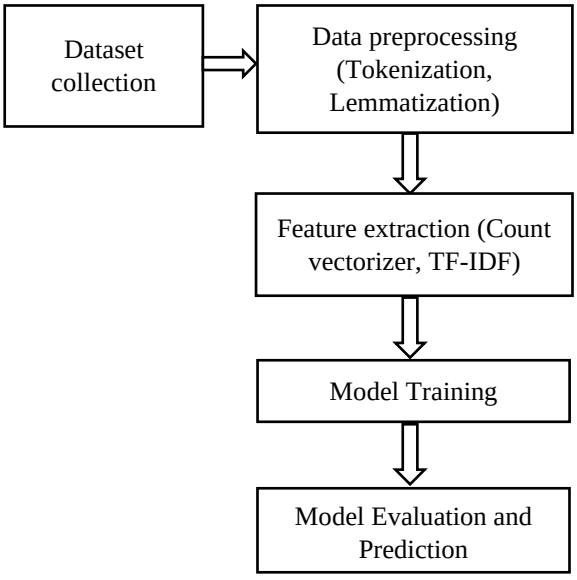

Fig.1. Workflow diagram of proposed Method

Subsequently, we employ five different classifiers to predict the class of the reviews. These classifiers include support vector

machine (SVM), linear support vector machines (LSVM), passive aggressive classifier (PA), logistic regression (LR), and multinomial naive Bayes (NB). The workflow of the deceptive review detection model is illustrated in the Fig.1.

4. RESULTS AND DISCUSSION

In this section, we evaluated our proposed approach on opinion spam dataset [1] and discussed about obtained results.

4.1 EXPERIEMNT OVERVIEW

For our study, we utilized a publicly available dataset which can be found on Kaggle under the title "Deceptive Opinion Spam Corpus". This dataset consists of 1600 reviews that have been classified into 800 truthful reviews and 800 fake reviews [1]. The reviews focus on the top twenty hotels in Chicago. The dataset was collected from two sources: TripAdvisor and Amazon Mechanical Turk.

In our analysis, we focused on two main attributes from the dataset: the review label and the review text. Other attributes were disregarded during the data pre-processing phase. After pre-processing the dataset, we transformed the textual data into vector representations using both count vectorizer and TF-IDF vectorizer. During feature extraction, we found that the choice of n-gram range and the maximum number of features played a crucial role in achieving higher accuracy. We observed that increasing the n-gram value had a noticeable impact on the overall accuracy of the models.

Additionally, we experimented with various values for the maximum number of features, ranging from 1000 to 50000. After evaluating different configurations, we found that setting the n-gram range to (1, 3) and the maximum number of features to 11000 yielded the best accuracy compared to other feature settings. Subsequently, we fed the vectorized data into our machine learning models. Table 1 provides a summary of the performance metrics achieved by all the machine learning models when using count vectorizer or bag of words (BOW) in combination with train-test split. Through these experiments, we aimed to optimize the feature extraction process and select the most effective configuration to enhance the accuracy of our models. Here we have used the below mentioned classifiers include support vector machine (SVM), linear support vector machines (LSVM), passive aggressive classifier (PA), logistic regression (LR), and multinomial naive Bayes (NB).

Table.1. Performance metrics of machine learning models using count vectorizer with train test split

Classifier	Feature	Performance Metrics			
		A	P	R	F
LR	BOW	**88.8%**	88	87	88
LSVM	BOW	87.8%	87	88	88
PA	BOW	88.3%	88	88	88
NB	BOW	88.4%	88	88	88
SVM	BOW	86.2%	86	86	87

The accuracy of five machine learning models using count vectorizer is outlined in Table 1. To evaluate the performance, we split the dataset into training and test sets using train-test split with a test size of 0.2. Notably, the linear support vector machine (LSVM) classifier achieved the highest accuracy of 91.8% when utilizing count vectorizer with a bigram model. Support vector machine (SVM) and logistic regression classifiers exhibited accuracy levels that were relatively close to LSVM. Conversely, the multinomial naive Bayes (NB) classifier yielded the lowest accuracy of 89.3%. To ensure comprehensive evaluation, we also incorporated k-fold cross-validation for dataset splitting. Specifically, we employed 5-fold cross-validation, where the dataset was divided into 80% for training and 20% for testing in each validation splitting round. This approach facilitated robust assessment of the models' performance and ensured reliable results.

Table.2. Performance metrics of machine learning models using count vectorizer with K Fold cross validation

Classifier	Feature	Performance Metrics			
		A	P	R	F
LR	BOW	90.3%	90	90	90
LSVM	BOW	**91.8%**	91	91	92
PA	BOW	90%	90	90	90
NB	BOW	89.3%	89	89	89
SVM	BOW	90.6%	90	90	91

The performance metrics of five machine learning models using count vectorizer with k-fold cross-validation is presented in Table 2. The dataset was split into training and test sets using k-fold cross-validation with k = 5, indicating that each fold represents a 20% test set and an 80% training set. Logistic regression achieved an accuracy of 88.8%, outperforming the other algorithms. Notably, the passive aggressive classifier and multinomial naive Bayes exhibited accuracy levels very close to logistic regression, indicating strong performance across the k-fold cross-validation. In terms of accuracy, support vector machine (SVM) and logistic regression classifiers demonstrated similar performance, while SVM exhibited the lowest accuracy among the algorithms evaluated.

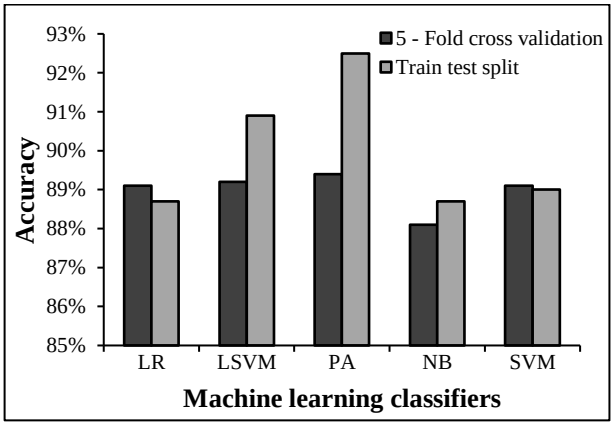

Fig.2. Performance of all machine learning models using count vectorizer

The Fig.2 presents a comprehensive overview of the performance of all five classifiers utilizing count vectorizer with both k-fold cross-validation and train-test split. The Fig.2 provides a clear visualization of the results, highlighting that the linear support vector machine (LSVM) classifier outperforms all other classifiers in terms of accuracy, achieving an impressive 91.8% accuracy. On the other hand, when employing 5-fold cross-validation, logistic regression demonstrates strong performance compared to the other classifiers. The Table.3 provides a summary of the performance metrics achieved by five machine learning models using TF-IDF vectorizer. As mentioned earlier, the models were evaluated using both train-test split and k-fold cross-validation. Remarkably, the passive aggressive classifier achieved the highest accuracy of 92.5% when utilizing TF-IDF vectorizer with a bigram model. The linear support vector machine (SVM) classifier also performed admirably, surpassing 90% accuracy. However, logistic regression and multinomial naive Bayes (NB) yielded slightly lower accuracies compared to the other classifiers.

Table.3. Performance metrics of machine learning models using TF-IDF vectorizer with train test split

Classifier	Feature	Performance Metrics			
		A	P	R	F
LR	TF-IDF	88.7%	88	87	88
LSVM	TF-IDF	90.9%	91	91	91
PA	TF-IDF	**92.5%**	92	93	93
NB	TF-IDF	88.7%	88	87	88
SVM	TF-IDF	89%	89	89	89

The Table.4 presents the performance metrics of five machine learning models using TF-IDF vectorizer with k-fold cross-validation. The passive aggressive classifier achieved an accuracy of 89.4%, slightly outperforming the other algorithms. Interestingly, the linear support vector machine (LSVM), support vector machine (SVM), and logistic regression classifiers exhibited accuracies that were relatively close to the passive aggressive classifier. Overall, our observations indicate that the passive aggressive classifier performed well on this dataset, delivering higher accuracy in both train-test split and k-fold cross-validation when employing TF-IDF vectorizer.

Table.4. Performance metrics of machine learning models using TF-IDF vectorizer with K Fold cross validation.

Classifier	Feature	Performance Metrics			
		A	P	R	F
LR	TF-IDF	89.1%	89	89	89
LSVM	TF-IDF	89.2%	89	89	89
PA	TF-IDF	**89.4%**	89	89	89
NB	TF-IDF	88.1%	88	87	88
SVM	TF-IDF	89.1%	89	89	89

Similar to the findings depicted in Fig.1, the linear support vector machine (LSVM) showcased superior performance when utilizing TF-IDF. Nonetheless, the highest accuracy of 92.5% was attained by the passive aggressive classifier. The Fig.3 provides a comprehensive overview of the performance of all classifiers when employing TF-IDF vectorizer in both train-test split and k-fold cross-validation settings. The Fig.3 effectively demonstrates that the passive aggressive classifier outshines all other classifiers in terms of accuracy, achieving an impressive 92.5% accuracy. Furthermore, the passive aggressive classifier exhibits strong performance across both train-test split and k-fold cross-validation when utilizing TF-IDF vectorizer.

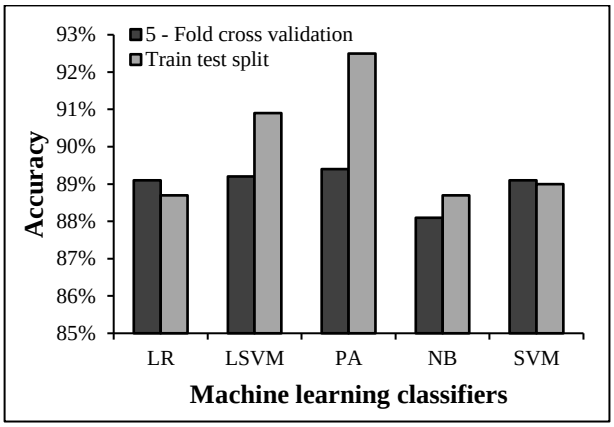

Fig.3. Performance of all machine learning models using TF-IDF vectorizer

4.2 EXPLAINABLE AI USING LIME AND SHAP

In order to enhance the transparency and interpretability of our proposed methods, we employed LIME and SHAP techniques to offer insights into the decision-making process of our model and to pinpoint the key features influencing its predictions.

The LIME-generated output comprises two main sections: (1) model prediction probabilities and (2) feature contributions to specific classes. Illustrated in Fig.10(a) are the LIME results for the truthful class, revealing that features “food”, ‘service”, and “definitely” played a significant role in predicting the deceptive class, while the remaining features primarily contributed to the truthful class.

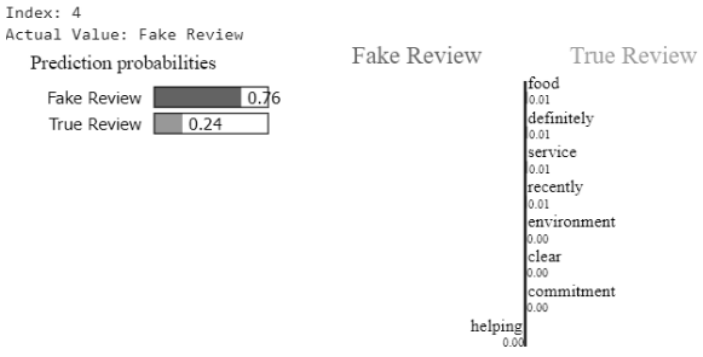

Fig.4. Sample output from LIME for each class for the deceptive dataset

The Fig.4 showcases a subset of the TF-IDF values for the top eight features identified by LIME, along with their respective classes. Notably, the terms “food”, “definitely”, and “service” are highlighted, signifying their potential significance in influencing predictions based on their TF-IDF values.

To ascertain the critical features across the entire dataset, we turned to the SHAP summary plot, which combines both feature importance and their effects on predictions, as depicted in Fig.5. Our observations revealed that features “hotel”, “food”, “night”, “floor”, and “experience” contributed most significantly to the model's predictions. Nevertheless, pinpointing the most influential features driving the model's overall predictions based solely on the SHAP summary plot remains a challenging endeavor.

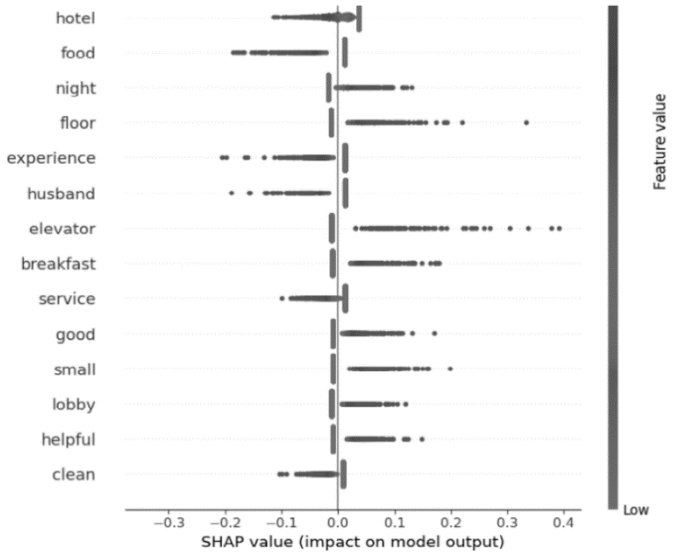

Fig.5. SHAP summary plot for the deceptive dataset

4.3 COMPARISON AND DISCUSSION

As previously mentioned, our model has shown superior performance compared to existing approaches. We conducted a comparative analysis between our model and previous works, which is summarized in Table 5. In order to detect deceptive reviews, we specifically compared our model to the work titled “Detecting opinion spams and fake news using text classification.” In the existing work, the authors employed stemming as a data preprocessing technique in their dataset [2]. They developed a fake review detection model that utilized n-gram features and TF-IDF metrics for text analysis. Their model achieved 90% accuracy using LSVM, slightly surpassing the 89% accuracy achieved by the original research study on the same dataset [1].

Table.5. Performance comparison of our model with existing works

Classifier	Feature	Accuracy	Research
SVM	Bigram	89%	[1]
LSVM	TF-IDF	90%	[2]
LSVM	Bag of words	**91.8%**	**Our results**
PA	TF-IDF	92.5%	**Our results**

In our study, using the same dataset, we utilized lemmatization for data cleaning. We explored both feature extraction techniques, namely count vectorizer and TF-IDF, and found that the choice of max features and n-gram range significantly impacted the accuracy. After evaluating different ranges, we selected max_features = 11000 and n_gram_range = (1, 3) to maximize the model accuracy. Our proposed model achieved 91.8% accuracy when using the linear support vector machine (LSVM) with count vectorizer, and 92.5% accuracy when using the passive aggressive classifier with TF-IDF vectorizer which is slightly higher than existing works [1]-[2].

5. CONCLUSION AND FUTURE DIRECTIONS

In recent years, the issue of opinion spam has gained significant attention due to the abundance of online-generated content. Fake reviews, in particular, have emerged as a major concern in the ecommerce industry and various social media platforms. Nowadays, it has become increasingly easy for anyone to post fake reviews on online websites. This deceptive practice is employed by many ecommerce businesses to mislead customers by posting positive reviews for certain products. Consequently, customers face challenges in distinguishing between reliable and unreliable products based solely on reviews.

To address this problem, our research paper focuses on detecting fake reviews by employing different feature extraction techniques. Initially, we explored various feature extraction methods commonly used by researchers in this field. Subsequently, we outlined traditional machine learning approaches for deceptive review detection, presenting summary tables and charts to summarize their performance. Moreover, we conducted a comparative analysis between existing works and our proposed approach for fake review detection. The results demonstrated that the passive aggressive classifier achieved the highest accuracy on the opinion spam dataset, surpassing the results obtained by this research study by 2.78% [2].

Despite the commendable performance demonstrated by our proposed model in detecting fake reviews, it is important to acknowledge certain limitations that shape the context of our findings. First and foremost, our reliance on a single dataset, while providing valuable insights, may restrict the model's applicability to other platforms and domains with distinct review characteristics. Additionally, the sensitivity of our model to feature extraction techniques and hyperparameters highlights the need for careful tuning and consideration of these factors in practice.

Furthermore, our comparative analysis primarily focused on a specific previous work, which may not fully encompass the breadth of existing approaches in the field of fake review detection. Moreover, our study did not extensively delve into the ethical implications and potential biases associated with fake review detection, crucial aspects when applying such models in real-world scenarios. Lastly, while our model's accuracy is promising, its practical deployment, scalability, and adaptability to the ever-evolving landscape of deceptive practices remain unexplored territories. These limitations, though important to recognize, do not diminish the value of our research but rather provide avenues for future work to enhance the robustness and applicability of our approach.

As we look ahead to the future, our commitment remains strong to advance our understanding of detecting deceptive reviews using deep learning neural networks. We plan to explore this subject further, considering it from various perspectives to

make our methods even more effective. One significant step is to gather more data to enhance the model's performance. We're also eager to dive into the world of advanced deep learning techniques like BERT, XLNET, RoBERTa, and word embeddings generated from GloVe. These methods offer exciting possibilities for improving our ability to spot fake reviews by capturing subtle language patterns and meanings. Furthermore, we'll adopt a method called feature selection. This technique will help us identify which parts of a review are the most important for detecting deception. By doing this, we'll make our models even more precise and better equipped to identify fake reviews. Our aim is to continue refining our approaches to make them more effective in identifying deceptive reviews in the future.

REFERENCES

[1] Jiwel Li, Myle Ott, Claire Cardie and Eduard Hovy, "Towards a General Rule for Identifying Deceptive Opinion Spam", *Proceedings of Annual Meeting of the Association for Computational Linguistics*, pp. 1566-1576, 2014.

[2] Ahmed Hadeer, Issa Traore and Sherif Saad**, "**Detecting Opinion Spams and Fake News using Text Classification", *Security and Privacy*, Vol. 1, No. 1, pp.1-9, 2018.

[3] Mukherjee Arjun, Bing Liu and Natalie Glance, "Spotting Fake Reviewer Groups in Consumer Reviews", *Proceedings of International Conference on World Wide Web*, pp. 191-200, 2012.

[4] Shojaee Somayeh, Masrah Azrifah Azmi Murad, Azreen Bin Azman, Nurfadhlina Mohd Sharef and Samaneh Nadali, "Detecting Deceptive Reviews using Lexical and Syntactic Features", *Proceedings of International Conference on Intelligent Systems Design and Applications*, pp. 53-58, 2013.

[5] P. Algur Siddu and S. Shivashankar, "Conceptual Level Similarity Measure based Review Spam Detection", *Proceedings of International Conference on Signal and Image Processing*, pp. 416-423, 2010.

[6] Y.K. Lau Raymond, Ron Chi-Wai Kwok, Kaiquan Xu, Yunqing Xia and Yuefeng Li, "Text Mining and Probabilistic Language Modeling for Online Review Spam Detection", *ACM Transactions on Management Information Systems*, Vol. 2, No. 4, pp. 1-30, 2012.

[7] Jindal Nitin and Bing Liu, "Opinion Spam and Analysis", *Proceedings of International Conference on Web Search and Data Mining*, pp. 219-230, 2008.

[8] Choi Wonil, Kyungmin Nam, Minwoo Park, Seoyi Yang, Sangyoon Hwang and Hayoung Oh, "Fake Review Identification and Utility Evaluation Model using Machine Learning", *Frontiers in Artificial Intelligence*, Vol. 5, pp. 1064371-1064378, 2023.

[9] Geli Fei, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malu Castellanos and Riddhiman Ghosh, "Exploiting Business in Reviews for Review Spammer Detection", *Proceedings of International AAAI Conference on Weblogs and Social Media*, pp. 175-184, 2013.

[10] Feng Song, Ritwik Banerjee and Yejin Choi, "Syntactic Stylometry for Deception Detection", *Proceedings of Annual Meeting of the Association for Computational Linguistics*, pp. 171-175, 2012.

[11] Jindal Nitin and Bing Liu, "Review Spam Detection", *Proceedings of International Conference on World Wide Web*, pp. 1189-1190, 2007.

[12] Li Fangtao Huang, Minlie Huang, Yi Yang and Xiaoyan Zhu, "Learning to Identify Review Spam", *Proceedings of 22nd International Joint Conference on Artificial Intelligence*, pp. 1-13, 2011.

[13] Li Jiwei, Claire Cardie and Sujian Li, "Topicspam: A Topic-Model based Approach for Spam Detection", *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 217-221, 2013.

[14] Myle Ott, Yejin Choi, Claire Cardie and Jeffrey T. Hancock, "Finding Deceptive Opinion Spam by any Stretch of the Imagination", *Proceedings of International Conference on World Wide Web*, pp. 1-8, 2011.

[15] Xie Sihong, Guan Wang, Shuyang Lin and Philip S. Yu, "Review Spam Detection via Temporal Pattern Discovery", *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 823-831, 2012.

[16] Mukherjee Arjun, Vivek Venkataraman, Bing Liu and Natalie Glance, "What Yelp Fake Review Filter Might be Doing?", *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 7, No. 1, pp. 409-418, 2013.