International Journal of Safety and Security Engineering
Vol. 15, No. 12, December, 2025, pp. 2519-2530
Journal homepage: http://iieta.org/journals/ijsse

Deep Learning Models for Fake Review Detection: A Focus on Bidirectional Encoder Representations from Transformers and Bidirectional Long Short-Term Memory

Wesam Hameed Asaad*, Ragheed Allami

College of Computer Science, University of Technology-Iraq, Baghdad 10066, Iraq

Corresponding Author Email: cs.20.48@grad.uotechnology.edu.iq

https://doi.org/10.18280/ijsse.151209

Received: 3 October 2025
Revised: 3 December 2025
Accepted: 17 December 2025
Available online: 31 December 2025

Keywords:
fake review detection, Bidirectional Encoder Representations from Transformers, Bidirectional Long Short-Term Memory, Yelp dataset, sentiment analysis, machine learning, rule-based methods

ABSTRACT

The fast rate at which users are creating such content in online review platforms has made deceptive reviews occur more often and create problems for both consumers and businesses. This paper explores deep learning (DL) and machine learning (ML) as fake reviews detectors, and includes a partial comparison with rule-based ones. We used the YelpZip dataset that has 608,598 reviews and 5,044 businesses, with 5,044 of these labeled as genuine and fake by Yelp's internal filtering. Two types of modeling pipelines were compared: traditional ML pipelines (Support Vector Machine (SVM), Extreme Gradient Boosting (XGBoost), Random Forest, and Logistic Regression) and DL pipelines (Bidirectional Long Short-Term Memory (BiLSTM), Bidirectional Encoder Representations from Transformers (BERT)). A simple rule-based system utilized the frequency of keywords and sentiment levels and was tested as a comparison. 80 percent of the data and the evaluation were carried out on the remaining 20 percent in all models. The best model of the models was BERT with an accuracy of 98% and an F1-score of 0.99. The BiLSTM model obtained 96 percent accuracy. Conventional models scored on average, and XGBoost, after having a 96 percent accuracy, had a lower recall. The findings reveal that contextualized embeddings are much more effective in detecting fake reviews as opposed to rule-based and conventional methods.

1. INTRODUCTION

Online shopping has increased the importance of user-created reviews in making people buy products. Nevertheless, this has also led to the appearance of opinion spam and fraudulent reviews that make it challenging to identify the presence of the deceitful material because of the slight language structures. Some of these methods of detection include rule-based filters and traditional machine learning (ML), as well as recent developments in deep learning (DL). Nevertheless, there are numerous sources that are not focused on generic datasets or the combination of sentiment features and contextual embeddings [1-4]. The research is structured to fill the gap in scalable and high-performing models applicable to an application such as Yelp and Amazon because of the lack of literature comparing extensive performances of various classifiers that involve sentiment analysis and extensive classifier assessments on large datasets. The originality of this work consists of (1) the combination of sentiment polarity characteristics and contextualized embeddings (Bidirectional Encoder Representations from Transformers (BERT) and Word2Vec) to optimize the accuracy of the model, and (2) comparison of various classifiers of the big YelpZip dataset using labeled reviews. This paper has shown that BERT and Bidirectional Long Short-Term Memory (BiLSTM) are much superior to rule-based and traditional classifiers when it comes to fake reviews detection. The paper is organized in the following way: Section 2 discusses literature related to it, and the methods and materials employed are described in Section 3. Section 4 explains fake review detection; the feature extraction techniques are stated in Section 5. Section 6 presents the ~~con~~ dropout. Proposed model in Section 7, feature extraction in Section 8, model training and evaluation in Section 9, confusion matrices in Section 10, and conclusions in Section 11.

The process gives companies the power to advertise their products and the ability to sabotage their competitors in terms of overall reviews generated, which results in people losing trust in online platforms. The identification of the fake reviews is carried out through an analysis of the review contents in order to differentiate between the authentic and the fake opinions based on the textual information. Machine-learning methods are applied to detect fraudulent reviews by detecting the specific or unusual use of words and phrase structures.

2. RELATED WORK

The problem has been addressed by detecting fake reviews with a rule-based system, behavioral analysis, traditional ML, and modern transformer-based DL models. All data and resource conditions have advantages of each type of method. Rule-based systems are based on some heuristics, which are developed by hand, like sentiment polarity mismatches or question keyword patterns. Although it is quick and easy to interpret, it cannot be generalized, and it can be affected by adversarial writing. As an example, Mohawesh [5] studied the case with keyword-based approaches, where the accuracy was lower with large and noisy datasets. This motivated us to have a rule-based baseline where sentiment thresholds were used and concentrated focus on more scalable models. Elmogy et al. [6] applied Naive Bayes (NB), K-Nearest Neighbors algorithm (KNN), and Support Vector Machine (SVM) algorithms on a real restaurant dataset to identify fake reviews and extract features from them, without using user behavior data.

Bansode and Birajdar [7] examined the user ratings of the hotel products based on the algorithmic sentiment analysis. Pal et al. [8] used DL along with opinion mining to detect the presence of fraud on reviews and found that both Convolutional Neural Network (CNN) and LSTM models detected fake reviews better. In a study conducted by Monica and Nagarathna [9], a sentiment analysis method was created based on real-time data of Twitter, and the researchers found that fake and authentic Twitter data utilized in sentiment analysis manifest abnormal sentiment patterns; its use is effective in detecting fake social media posts.

Behavioral models examine the activity pattern of the users, like the review frequency, tendency to rate reviews, or the time stamp of the reviews. Wang and Wu [10] proved that behavioral signals add to the detection performance. However, our work actively removes information that is textual to test the capability of the models without textual information, which is typically not feasible in real-time applications using user metadata.

3. METHODS AND MATERIALS

This paper will perform a comparative appraisal between different ML and DL architectures in fake review detection, and classical classifiers (SVM, Random Forest, Extreme Gradient Boosting (XGBoost), Logistic Regression), and deep architectures (BiLSTM, BERT). It is based on a large reviewed dataset, which has labels and a simple rule-based classifier to use as a baseline. This study is a single pipeline of detection with text preprocessing, feature extraction, model training, and evaluation of the performance that may not rely on the behavioral metadata to make timely deployments to real-time systems, merely because of textual review content.

3.1 Techniques of fake reviews detection

The authors of the fake reviews, or the spammers as they are called, are the ones who write misleading or spam reviews, and these reviews may be either positive or negative in terms of content, depending on their motive. The dilemma of finding fake reviews consists of the ability to classify them into true and fake, which can be solved with the help of Natural Language Processing (NLP), ML algorithms, or graph networks. Different models were tested on the data of such platforms as Yelp, TripAdvisor, Amazon, and YouTube. Since fraudsters are changing their strategies persistently, the use of online reviews is still an important source of consumer information, thereby necessitating the use of sound detection techniques [11-13].

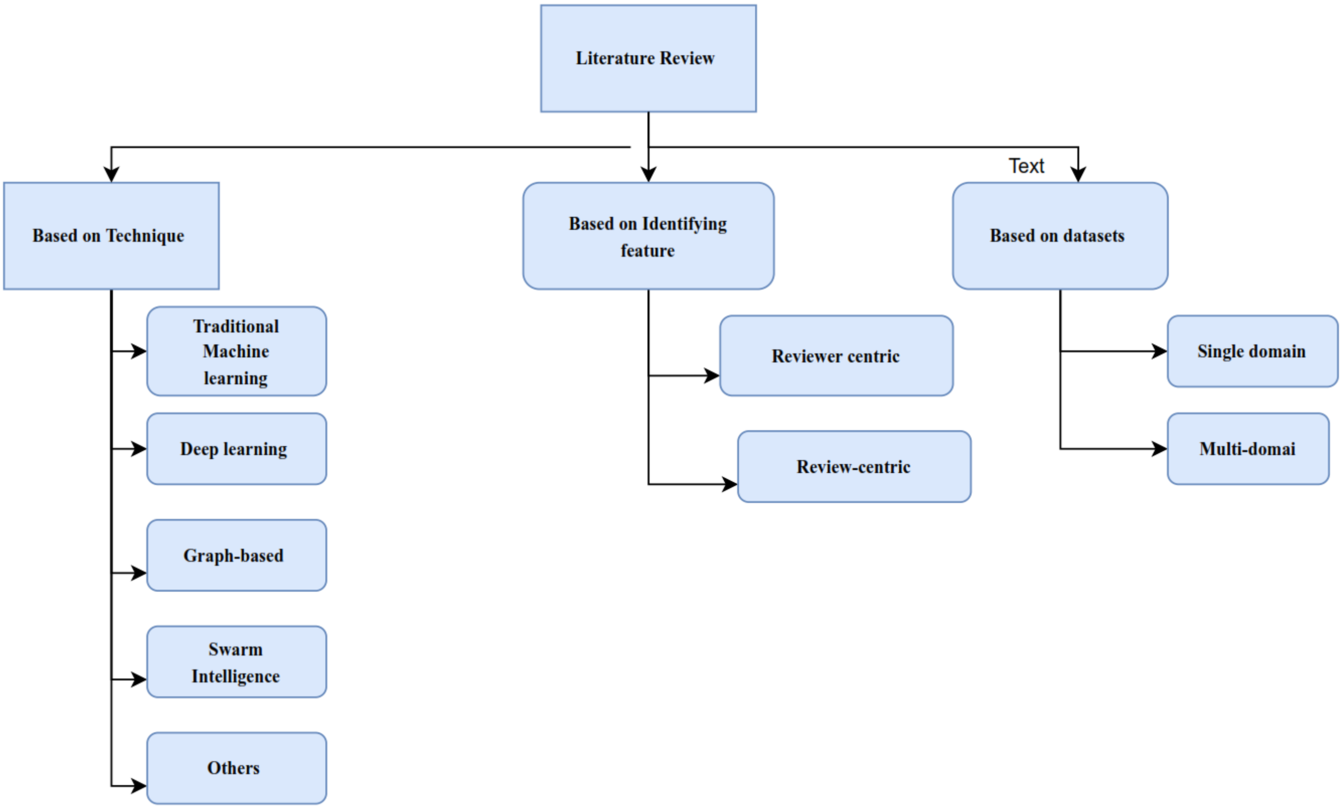


Figure 1. Workflow overview [14]

To identify fake reviews with the help of AI, a model must pass through a few processes, as shown in Figure 1. 1) Data Collection: A model receives real and false reviews on websites such as e-commerce and social media. 2) Feature

Extraction lays emphasis on such critical elements of reviews as the text, user profile, sentiment, and patterns of language. 3) Data Preprocessing enhances the quality of the input by correcting text and eliminating noise. 4) The reviews are labeled to identify the authentic and fake ones, which is done manually and automatically. 5) Model Training applies different algorithms, such as Logistic Regression and DL models, to acquire patterns that represent genuine reviews. 6) Model Evaluation models the performance with regard to accuracy and other measures. 7) Model Deployment enables the detection of fake reviews in real time. 8) Continuous Learning modifies the model to reflect the new patterns and feedback to enhance performance. Many algorithms are applied and are still being developed; they are rule-based, ML, or DL to be more effective in detecting fake reviews.

Rule-Based Detection Baseline: This was done via a straightforward rule-based model to create a non-learning mark. The rules were developed on the basis of sentiment mismatch, over polarity, and keyword indicators as informed by the literature in the past [5]. The applied rules include:

• Rule 1 (Polarity mismatch): In case a review is rated with a polarity score of 0.9 or less or 1.0 or more and its star rating is between 2 and 3, it is considered to be fake.

• Rule 2 (Length threshold): all reviews of less than 10 words and those of more than 500 words are considered suspicious.

• Rule 3 (Keyword pattern): The reviews that include over two of the keywords in the list below are considered to be flagged as such:

The rule-based classifier generates a binary debt label (0 = fake, 1 = real) by majority voting between rules. This benchmark was tested on the 20 percent test split that ML models were tested on.

3.2 Support Vector Machine

The classifier assists with both linear and nonlinear scenarios of the supervised ML classification algorithm. The SVM categorizes the data into classes and then finds the hyperplane, which separates the data into groups. The primary concept of SVM in sentiment classification is determining the hyperplane [15]. Consider Figure 2, which has two distinct categories classified by decision boundaries or hyperplanes.

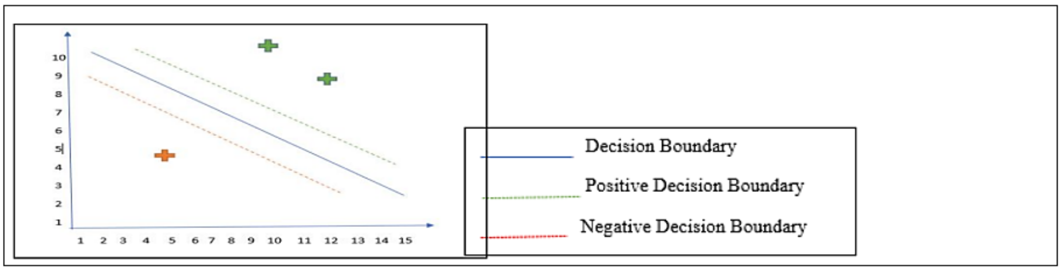


Figure 2. Support Vector Machine (SVM) hyperplane [6]

3.3 Random Forest

Random Forest is a type of ensemble classifier that consists of several decision trees, and all of these trees are built in a random way, that is, features are chosen randomly to split nodes. The approach improves the flexibility to high dimensionality data by modifying the hyperparameters, and classifying well. It is normally used in classification and regression activities, and the group decision trees are well-informed and dependable in their predictions [3, 10].

3.4 Logistic Regression

Logistic Regression [6] is another straightforward supervised algorithm for classification. It relies on locating a hyperplane that classifies the data. The LR classifier effectively categorizes reviews into polarity groups by employing both training and testing datasets. Logistic Regression, the swiftest predictive classifier, circumvents overfitting and gives the best generalization [15]. The LR demonstrates improved performance on the new dataset.

3.5 Extreme Gradient Boosting classifier

The XGBoost algorithm, created at the University of Washington by Tianqi Chen and Carlos Guestrin, is an ML method based on ensembles, which is highly efficient in terms of time consumption and memory, as well as in managing imbalanced data sets. Designing around a gradient boosting model using decision trees, XGBoost is an efficient method to enhance the performance and stability of models, and therefore, a useful resource to the distributed ML community [16, 17].

3.6 Bidirectional Long Short-Term Memory

Conceptual Modification (BiLSTM) is the adjustment of an LSTM framework that handles training in both forward and backward time. A modified LSTM network enables one to scan input in both the forward and backward directions at the same time. The two layers in BiLSTM are arranged in two directions: forward and backward. The product of such a network will be the combination of the products of these two layers [18]. This technique is also useful in BERTConvNet to detect fake reviews and achieve positive results on video review data sets [19]. Unsupervised clustering methods such as GrFrauder have also become an avenue to be looked upon in group spam reviewer detection [20].

3.7 Bidirectional Encoder Representations from Transformers

BERT is a pre-trained DL model used for sentiment analysis and language translation. The context of sentences or phrases is effectively used, which is particularly important for identifying fake reviews. It uses contextual understanding and accurately determines whether a review is genuine or fake. An encoder-only model, BERT, concentrates on grasping the input data by encoding its contextual information, rather than generating new material [21, 22].

4. OPINION MINING-BASED FAKE REVIEW DETECTION

This paper is an opinion mining on detecting fake reviews using a DL strategy. It combines several lexicons, a Word2Vec model, and a sentiment analysis attention mechanism (AM). It is an AM that creates sentiment data with the help of SentiWordNet and the lexicon described by Pal et al. [8] and maintains text coherence. The first model in the analysis is the Word2Vec model, which helps to investigate the sentiment features and explain the sentiment conflict in context, similar to how a human being perceives the sentiment.

4.1 Text pre-processing

The first step of NLP is text preprocessing, which seeks to transform the raw text into a structured form that is easily processed by the machine learner. It is important because it has a direct effect on the quality of classification outputs by eliminating noise and variations, and redundant information. Some of the techniques applied are case normalization to lowercase, removal of stopwords, and the lemmatization technique, which is essential in sentiment analysis. The preprocessing process entails 4 stages, namely, text normalization, tokenization, lemmatizing and removal of stop words [23].

4.2 Text normalization

Elimination of punctuation in NLP eliminates redundant tokens that enhance the data volumes and processing costs. To remove the inconsistencies, cleaning and normalization of the text will be necessary, which will allow creating the features and ranking the text efficiently with the lowest number of irrelevant noises due to inactive characters and markup tags [24].

4.3 Tokenization

Tokenization breaks down raw text into smaller units known as tokens. These tokens aid in understanding the context when developing the model for NLP. Tokenization enables us to interpret the meaning of the text by analyzing the sequence of the words. Tokenization is commonly understood as any type of natural language text preprocessing. Tokenization replaces sensitive data with a unique code while keeping the original information intact [25].

4.4 Lemmatization

Employing this method, the text is converted into its stemmed format. Recent approaches in literature often use a clever strategy to combine diverse word patterns by attempting to eliminate affixes. Suffixes ending with es or -s can convert a noun to singular or plural. To get a verb in its present or past participle, utilize the suffixes -ing or -ed. The -est suffix enables adjectives to take comparative or superlative forms. Searching for a specific word like "sunsets" in SentiWordNet can be difficult because the algorithms produce incorrect stemmed forms [8, 26].

4.5 Stop word removal

One of the most common preprocessing mechanisms applied to applications of NLP comprises the elimination of stopwords. The point is to eliminate the words common to all the documents [16]. Common words like conjunctions (or, and, but) and pronouns (he, she, it) should be removed because they provide little to no value in the classification process. Each feature should be eliminated if it matches any stop words.

5. FEATURE EXTRACTION TECHNIQUES

Feature extraction methods (FET) are crucial in sentiment analysis because they help identify human opinions, attitudes, and behaviors based on posts, comments, etc. Many vital features have been extracted from the chosen text dataset. It came from social media information for sentiment analysis in the word-based features. The key features are part of three effective sentiment analysis techniques: term frequency-inverse document frequency (TF-IDF), word embedding, Word2Vec, and BERT embedding [27]. These methods extract advantageous features. FET for text datasets is implemented using Python programming.

5.1 Term frequency-inverse document frequency

TF-IDF is an acronym that represents Term Frequency Inverse Document Frequency. TF-IDF is a powerful feature extraction technique that identifies and emphasizes the most relevant characteristics within a dataset through weighted scoring. It assigns weights to each phrase in the text documents to better restructure the performance of the trained model [28].

5.2 Embedding of attention word

One of the key developments in the study of DL is AM. The strategy involves the use of lexicons and AM to explain to the students the point concerning vitality in sentence word recognition vested in sentiment scores. The proposed solution takes word representations either in the form of semantics or sentiment by utilizing lexicons and Word2Vec models. By relying on the SentiWordNet and the lexicon developed by Liu, we extract useful sentiment information of words. The method of calculating the sentiment scores is applied to extract the sentiment information of the lexicons [8].

6. DROPOUT

It was proposed to employ dropout as a regularization technique for neural network (NN) classifiers. During training, dropout strategically excludes certain neurons, ensuring they are not activated. During the forward phase, earlier ones do not influence the activation of later neurons, and weight updates do not occur for the neurons in the backpropagation step. The most frequent method of addressing overfitting is dropout. The suggested approach is also helpful for handling low-priority levels.

Dense layer. The system classifies output into positive and negative sentiments using a fully connected dense layer with a SoftMax function.

7. PROPOSED MODEL

Figure 3 shows the structure's main components, illustrating how to split data into training and testing sets. Afterward, the trained model was evaluated using the test data. This paper compares various machine-learning models to BERT and BiLSTM for sentiment analysis of our fake reviews dataset.

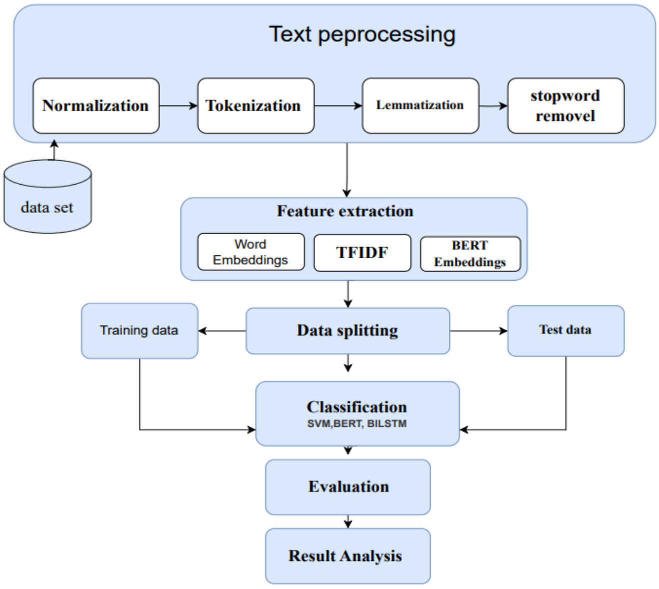


Figure 3. Generalized framework of the proposed approach

7.1 Dataset

We analyze the YelpZip data [28], which consists of about 608,598 reviews of 5,044 restaurants, left by 260,277 reviewers, and noted by the Yelp anti-fraud filter as either authentic or as a fake review. So, taking this labeled data as a basis of our tests, we build on the previous research conducted by Salminen et al. [29] by applying two sophisticated language models, BiLSTM and BERT. BiLSTM produces both forward and backward word dependencies of the reviews, whereas BERT uses self-attention to provide token representations that are dependent on the context. The results that we have found show that the application of these models also increases our capability of differentiating between fake and real reviews, as they are able to capture the syntactic as well as semantics of the language.

All reviews presented in the dataset contain the following fields, namely review text, star rating, review date, user id, business id, and a score indicating its authenticity (0 0 fake, 1 real) depending on the Yelp algorithm. The dataset is disproportionate as it has 72,336 counterfeit reviews (11.9 Percent) and 536,262 authentic reviews (88.1 Percent). Class proportions were also preserved, and an 80 /20 train-test split was employed. Two model setups were compared: text-only models (TF-IDF, Word2Vec, and BERT embeddings) and text and metadata models, which included numeric variables such as the length of review, the star rating, and sentiment polarity. To prevent the leakage of the data and concentrate on the content-driven performance of the detection, the user and the product identifiers were removed. It involved lower-case, punctuations and stopword removal, lemmatization, and contraction expansion preprocessing, and sentiment scores were calculated with TextBlob in certain settings.

7.2 Text preprocessing

Text reviews are pre-trained in critical preparation phases before ML models are trained to improve the quality of input data and the model performance. With LSTM networks, words are represented as numbered tokens, and then sequences are simulated to be of the same length. The conversion into tensors and separation into training and validation sets are then carried out with this set of sequences. On the other hand, BERT preparation includes tokenization by wordpiece, applying [CLS], [SEP], and adding attention masks to identify useful tokens and the tokens that are used as padding, thus improving the learning effect.

7.3 Expanding contractions

The short forms of words were replaced with their regular forms to keep the message clear. As an example: “won’t” changes to “will not” and “can’t” becomes “cannot”. By completing grammar checks, we secure the text’s structure, which helps the next steps in text processing.

8. FEATURE EXTRACTION

Feature extraction is a basic part of developing pattern recognition and ML models. The goal is to improve model performance by retaining the most useful data and discarding the unnecessary parts. Because of this process, unstructured text is changed into forms suitable for DL and ML classification.

Rule-Based Performance: These results gave the rule-based model Accuracy: 72.1%, Precision: 70.4%, Recall: 66.8% and an F1-score 68.5% on the test set. Although it was fast and easily interpretable, it was far worse than all ML models, particularly borderline or subtle reviews. These findings again confirm the inadequacy of strict sets of rules in high-variance, user-generated content.

8.1 Term frequency–inverse document frequency

NLP entails transforming text into a format a machine-learning model can grasp: numbers. This can be achieved using a vectorizer, which generates a vector or matrix that can be input into the model. The chief kinds of vectorizers are a count vectorizer and the TF-IDF vectorizer. The count vectorizer makes a matrix displaying how often each word occurs in a document. The issue with a count vectorizer is that it emphasizes common words like ‘a,’ ‘and’, and ‘of’. Another drawback is that it does not consider the other words in a document or review. A brief review with a specific word can carry more weight than a longer review. This is where the TF-IDF vectorizer steps in. TF-IDF assesses a word's distinctiveness and significance by comparing how often it emerges in a review to how many reviews feature that word.

8.2 Word embeddings (Word2Vec)

Representations of vectors that continuously store similarities between words in their context.

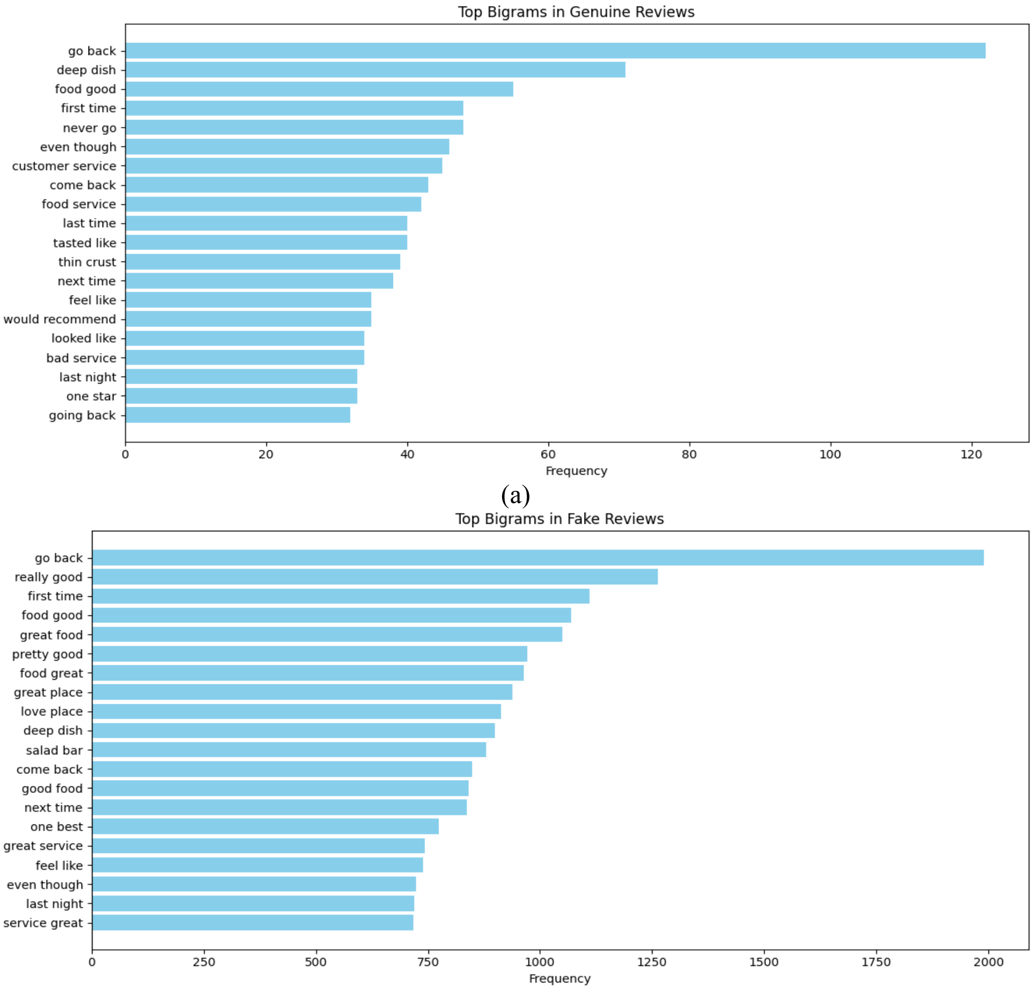
8.3 N-gram features

Sequential n-grams (e.g., bigrams and trigrams) were generated to capture local patterns and dependencies between adjacent words. These patterns help detect stylistic inconsistencies that may signal deceptive intent. This project analyzed text reviews and found a direct correlation between bigrams and trigrams (two- and three-word phrases) and review ratings, such as 1-star and 5-star ratings, as illustrated in Figure 4(a), Figure 4(b), Figure 5(a), and Figure 5(b).

The term frequency in both fraudulent and legitimate reviews was well brought out by the word cloud visualization. Legitimate reviews usually have words such as high quality, easy to use, and very satisfied, whereas the words amazing, the best, and perfect are used in fraudulent reviews without any details that justify their use and make them look less credible, see Figures 6-8.

8.4 Sentiment features

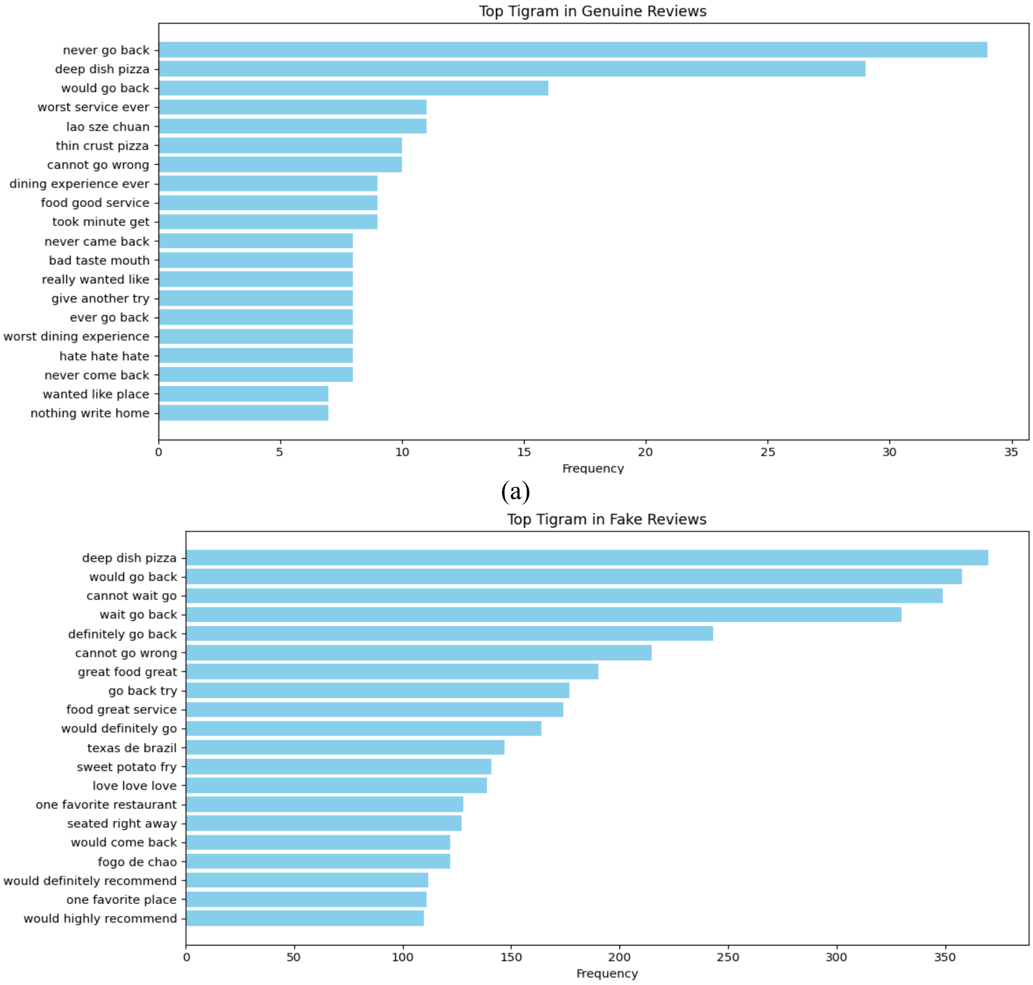
The TextBlob library will provide a rating of -1 to 1 as the sentiment score of the review, allowing one to detect the difference between the sentiment polarity rating and the rating of the stars, providing evidence of the fraudulent character. Only to simplify the calculations, the scores of polarities are converted to a binary sentiment label to help the model identify the use of emotionally charged language that may suggest deception. The sentiments are categorized into positive, negative, and neutral, with refinements by the use of word embeddings to represent reviews better. The dataset was long and made it possible to analyze different aspects of products and different ends where sentiment is concerned, to measure how successful the binary classification approach has been. In addition, a combination of sentiment-based, semantic, and behavioral data was beneficial in the fraudulent review detection, and this confirms the significance of a bookriddles-papsseras type of data in deriving a fraudulent rating.



(a)

(b)

Figure 4. (a) Top bigram in genuine reviews; (b) Top bigram in fake reviews



(a)

(b)

Figure 5. (a) Top tigram in genuine reviews; (b) Top tigram in fake reviews

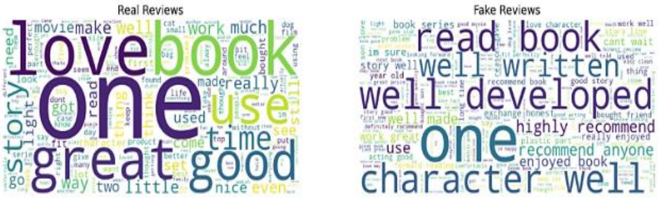


Figure 6. Word cloud

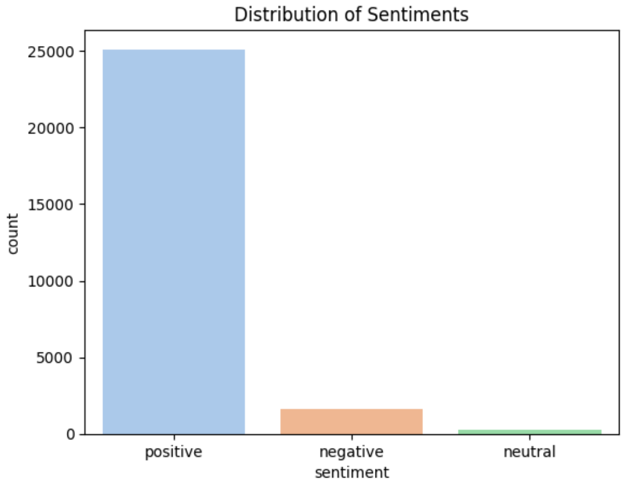


Figure 7. Distribution of sentiment

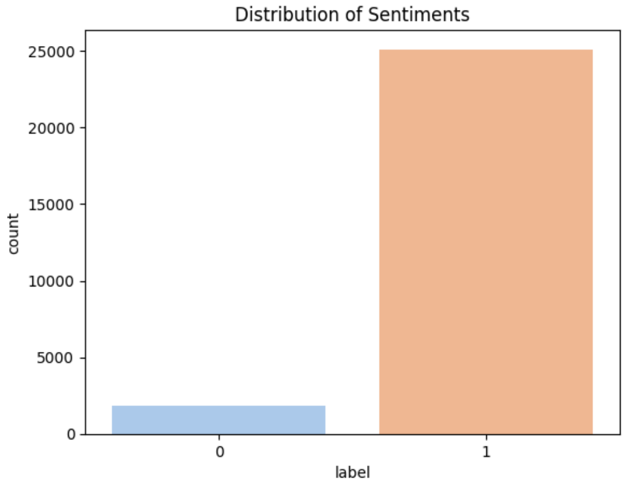


Figure 8. Sentiment label

8.5 Word embeddings in Bidirectional Long Short-Term Memory

In the BiLSTM sentiment classification pipeline, words are converted into high-dimensional embeddings that capture important semantic and syntactic relationships. Vector representations provide embeddings that help the model understand individual words and their context. A BiLSTM network enables the model to learn both forward and backward dependencies in text, enhancing sentiment interpretation.

8.6 Contextual embeddings (Bidirectional Encoder Representations from Transformer)

This analysis considers BERT embeddings. BERT is a computational approach that converts words into numbers. ML models necessitate numerical inputs instead of text. An algorithm that converts words to numbers enables the training of these models using original text data. As such, language is better understood than static embeddings, resulting in superior outcomes in detecting fake reviews.

9. MODEL TRAINING AND EVALUATION

As in the case of the base experiment(s), the Amazon product reviews became a part of the dataset, resulting in a bigger training and evaluation corpus. The tokenization, the removal of stop words, lemmatization, and extraction of features were performed on the dataset. Each of the reviews was noted as fake or true. The training pipeline was provided with sentiment by adding polarity scores by TextBlob, which were used as both continuous features and labels of sentiment. Different models, such as Logistic Regression, the Random Forest, SVM, XGBoost, BERT embeddings, and BiLSTM networks, were trained and tested using the dataset. Findings demonstrated that adding sentiment-based variables to the process of detecting fake reviews was a considerable help, and the BiLSTM network, as well as the XGBoost, captured the contextual meaning and strengthened the classification, respectively. Generally, sentiment analysis came in handy in maximizing the reliability of fake review systems.

Further analyses are found in comparative Tables 1-5 that compare the results of the proposed system with those from previous studies. This comparison demonstrates that the proposed solution improves how fake reviews are detected.

The proposed system showed high accuracy, achieving a notable 98% in distinguishing fake reviews from genuine ones. The system performed well across all areas, demonstrating its ability to handle various datasets compared to other advanced models. Tests revealed that our suggested system was more accurate than several other recent studies, especially with imbalanced data.

Table 1. A classifier using term frequency-inverse document frequency (TF-IDF) feature

Class	Classifier	Precision	Recall	Accuracy	F1-Score
0	Support Vector Machine (SVM)	0.79	0.45	0.96	0.57
1		0.96	0.99		0.98
0	Extreme Gradient Boosting (XGBoost)	0.76	0.58	0.96	0.66
1		0.97	0.99		0.98
0	Random Forest	0.83	0.27	0.95	0.36
1		0.95	1.00		0.97
0	Logistic Regression	0.80	0.22	0.95	0.41
1		0.95	1.00		0.97

Note: Class 0 = Fake reviews, Class 1 = Genuine reviews

Table 2. A classifier using Word2Vec feature

Class	Classifier	Precision	Recall	Accuracy	F1-Score
0	Support Vector Machine (SVM)	1.00	0.01	0.94	0.01
1		0.94	1.00		0.97
0	Extreme Gradient Boosting (XGBoost)	0.13	0.01	0.93	0.02
1		0.94	1.00		0.97
0	Random Forest	0.62	0.01	0.94	0.03
1		0.94	1.00		0.97
0	Logistic Regression	0.50	0.01	0.94	0.01
1		0.94	1.00		0.97

Note: Class 0 = Fake reviews, Class 1 = Genuine reviews.

Table 3. A classifier using Bidirectional Encoder Representations from Transformers (BERT)

Class	Classifier	Precision	Recall	Accuracy	F1-Score
0	BERT	0.82	0.81	0.98	0.81
1		0.99	0.99		0.99

Note: Class 0 = Fake reviews, Class 1 = Genuine reviews.

Table 4. A classifier using Bidirectional Long Short-Term Memory (BiLSTM)

Class	Classifier	Precision	Recall	Accuracy	F1-Score
0	BiLSTM	0.74	0.50	0.96	0.60
1		0.97	0.99		0.98

Note: Class 0 = Fake reviews, Class 1 = Genuine reviews.

Table 5. A comparison of the accuracy and sentiment integration

Class	Classifier	Precision	Recall	Accuracy	F1-Score
Fake	Extreme Gradient Boosting (XGBoost)	0.85	0.75	0.81	0..80
Real		0.78	0.78		0.82
Fake	Random Forest	0.76	0.74	0.75	0.75
Real		0.74	0.76		0.75
Fake	Bidirectional Long Short-Term Memory (BiLSTM)	0.97	0.90	0.94	0.93
Real		0.91	0.97		0.94
Fake	XGBoost	0.75	0.76	0.75	0.75
Real		0.75	0.75		0.75
Fake	Random Forest	0.76	0.74	0.75	0.75
Real		0.74	0.76		0.75
Fake	BiLSTM	0.86	0.87	0.87	0.87
Real		0.87	0.86		0.86

The combination of traditional ML with modern techniques such as BERT and BiLSTM helped improve the model's ability to work well on data from different sources, as can be seen in Table 6.

Table 6. Comparison of selected researchers

Authors	Methodology	Classifier	Accuracy (%)	Dataset	Notable Findings
Mohawesh et al. [4]	Feature extraction	BERT, RoBERTa, and XLNet	86.20%	Yelp and Amazon	Contextual embeddings improve detection accuracy
Mohawesh [5]	Feature engineering	NB, SVM, BiLSTM, CNN, DNN	BiLSTM was 77.59; CNN, DNN was 66.77%, 79.64%	Yelp Zip	RoBERTa 91.2%
Elmogy et al. [6]	Features extracted	NB, SVM, Logistic Regression, Random Forest, KNN	82.40%	Yelp	The F-score has increased by 3.80%, when considering the extracted reviewers' behavioral features
Pal et al. [8]	Features extracted	LSTM	92%	Amazon	Word, semantic sentiment
Proposed model	Feature extraction	SVM, Random Forest, BERT, BiLSTM, Logistic, XGBoost	0.98	Yelp	Word, semantic sentiment

Note: BERT = Bidirectional Encoder Representations from Transformers; NB = Naive Bayes; SVM = Support Vector Machine; BiLSTM = Bidirectional Long Short-Term Memory; CNN = Convolutional Neural Network; DNN = Deep Neural Network; KNN = K-Nearest Neighbors; XGBoost = Extreme Gradient Boosting.

10. CONFUSION MATRICES

The confusion matrices are being used to evaluate, through visual analysis, the performance of the classification models that were trained using TF-IDF features, namely in separating the valid and false reviews. These models are also tested by the use of key performance measures like accuracy, precision, recall, and F1-score. Sentiment analysis, which assigns binary labels and polarity scores, can be greatly used to improve the model on deceptive language detection, and it generates better accuracy and F1-scores. Such a mixture of methods can provide a more effective way to recognize dishonest content, and the result would be the growth of confidence in the online reviews by the user, see Figures 9 and 10.

Semantic and syntactic relationships between review words were captured using Word2Vec embeddings. Word2Vec represents words as dense vectors in a continuous space, placing similar words close together. By averaging all the embeddings in a review, we obtained a fixed-length vector for the classification models. This approach helps the model distinguish between fake and genuine reviews based on language patterns, as shown in Figure 11.

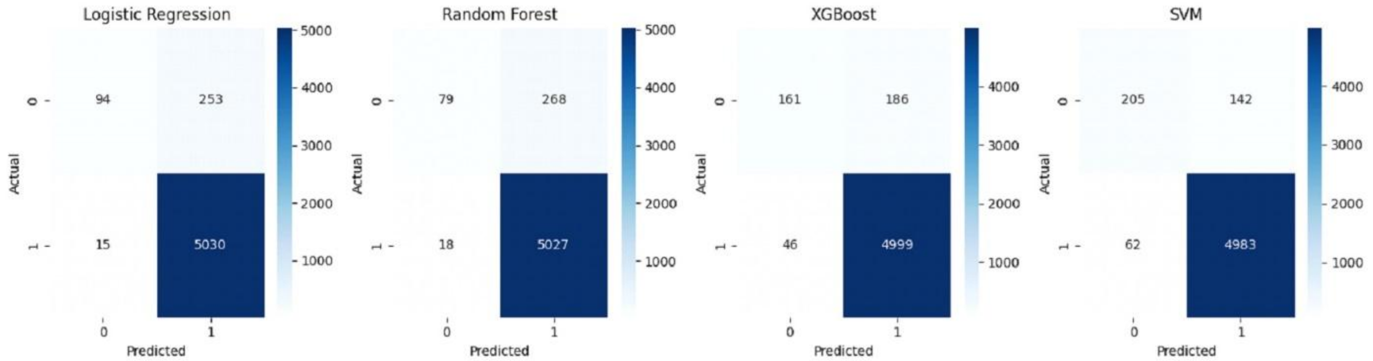


Figure 9. Confusion matrix sentiment feature

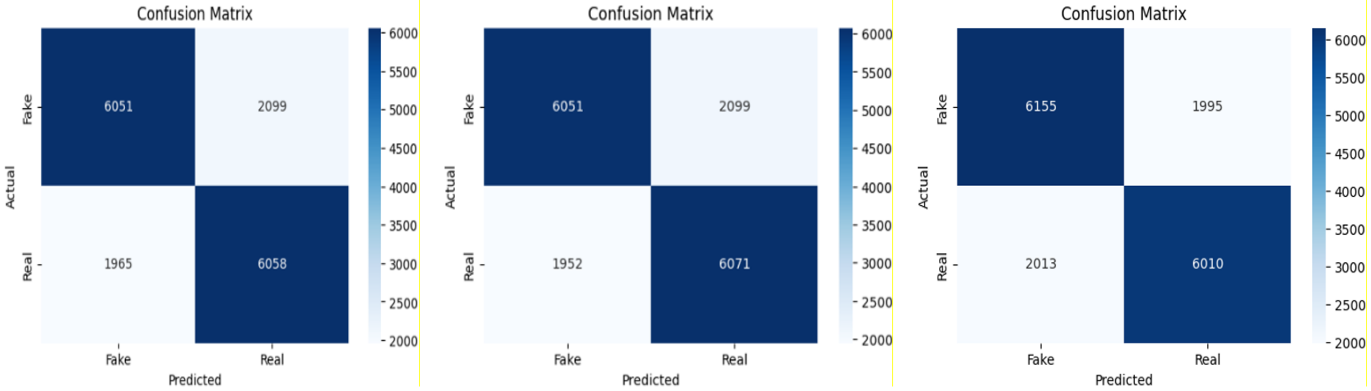


Figure 10. Confusion matrix sentiment integration

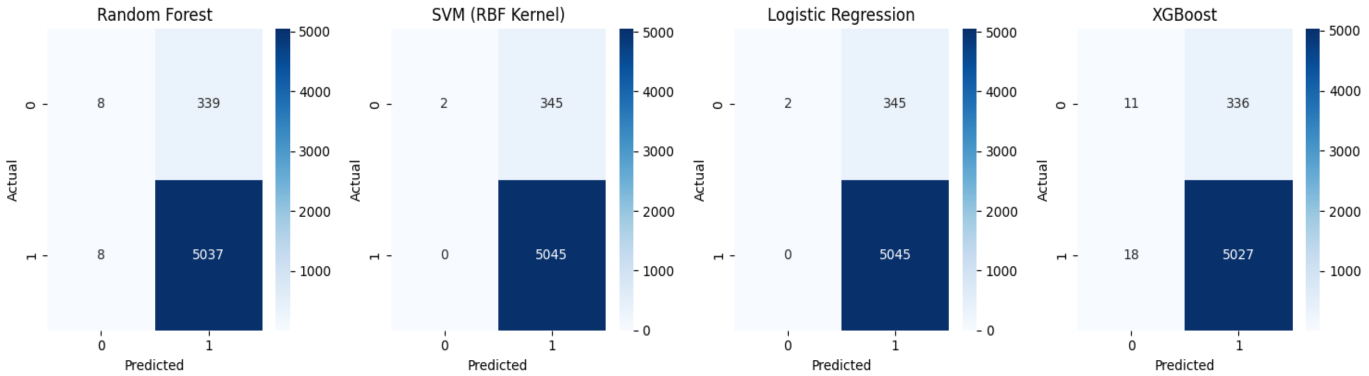


Figure 11. Confusion matrix Word2Vec feature

In this study, BiLSTM networks and BERT were used to enhance the detection of fake reviews. BiLSTM, which belongs to the RNN family, features long-range relationships between sequential information. BERT also uses a transformer model to gain a subtle insight into text. The models were trained individually or in a hybrid approach by using BERT representation to classify using the LSTM or fully connected networks. As performance metrics, it had accuracy, precision, and recall, as shown in Figures 12 and 13.

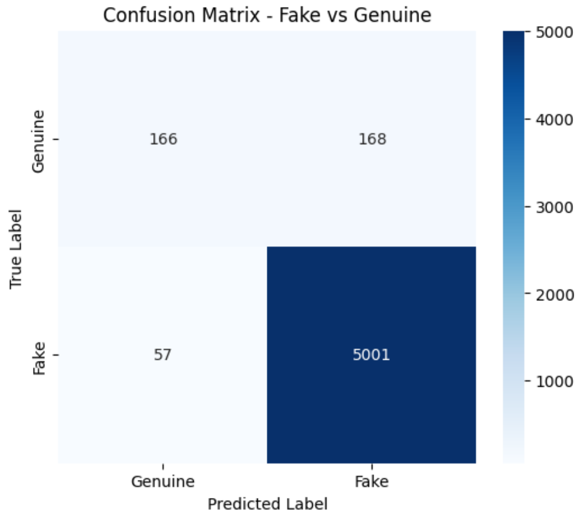


Figure 12. Confusion matrix Bidirectional Long Short-Term Memory (BiLSTM)

In all the models, the performance by the indicator of accuracy did not show consistency because of an imbalance in the dataset classes (fake: 12 percent, genuine: 88 percent). Interestingly, Word2Vec-based models produced high average accuracy (around 94.00) at the cost of very low recall of the minority group, and SVM and XGBoost produced almost 0.01 recall of fake reviews. Due to this, macro-averaged F1-scores and per-class recall represent better information measures. BERT and BiLSTM models were the most successful, reaching the class 0 recall of more than 80% and macro-F1 of more than 0.90, and preserving high precision. Rule-based detection, on the other hand, showed inferior recall and F1-scores compared to learning-based methods, suggesting its weakness in complex cases.

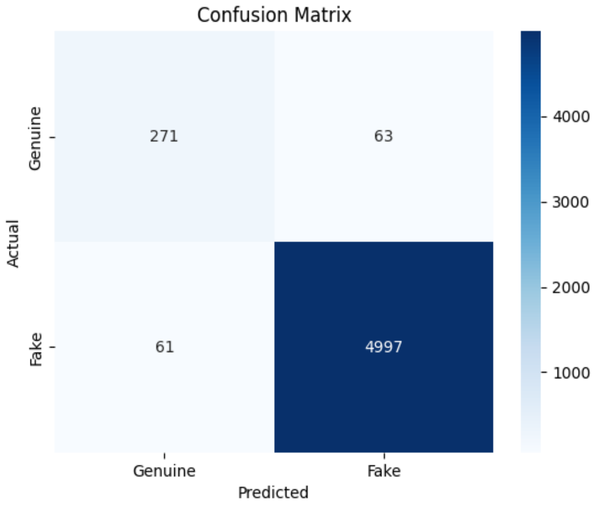


Figure 13. Confusion matrix Bidirectional Encoder Representations from Transformers (BERT)

BiLSTM models, too, were very successful in competing with others by capturing sequential text dependencies. Those based on rules were less successful, as they provided transparency but were bad in terms of adaptation and awareness of contexts. A trade-off between interpretability and performance has been identified; e.g., models such as Logistic Regression are easy to understand but do not capture the ability of the model to capture complex patterns, and thus give poorer results than transformer-based models. Overall, both BERT and BiLSTM were better in both fake review detection and false positive and negative balance in case it is evaluated with the help of the same metrics over the same dataset.

11. CONCLUSIONS

The paper involved a comparative analysis between rule-based, conventional ML, and the deep perceptions that identify fake reviews against the YelpZip dataset. Findings showed that contextual embedding based (BERT) and sequence modeling (BiLSTM) models outperformed classical models and rule-based baselines, particularly in class imbalance conditions of detecting fake reviews. The most successful model (BERT) had a high brand accuracy (98%), and both classes performed well in terms of macro-F1 and recall, thereby qualifying to be applied in review moderation with high stakes.

It was also noted that the standard models based on TF-IDF and Word2Vec features were limited because they could not detect minor classes even after showing good overall performance. Rule methods, however interpretable, did not have the flexibility of handling deceptive patterns in language in real-life data.

Future work will focus on:

• Adding metadata properties, like date of review, length of review, and the star rating, as features in order to enhance feature representation.

• Investigating user-level or product level signals, e.g., reviewers' history or burst, to enhance fake group.

• Carrying out strength tests on various datasets (e.g., Amazon, TripAdvisor) to test generalization.

• Incorporating explainable AI methods (or SHAP, or LIME) to increase model transparency in high-impact users of AI.

Through our content-based characteristics in the given work, we would give a scalable and dependable foundation of fake review postings, and we would give a clear roadmap of how this detection system could be improved in the future by the incorporation of hybrid algorithms to integrate semantic, behavioral, and temporal indicators.

REFERENCES

[1] Abd, M.J., Hussein, M.H. (2024). Fake reviews detection in e-commerce using machine learning techniques: A comparative survey. BIO Web of Conferences, 97: 00099. https://doi.org/10.1051/bioconf/20249700099

[2] Yadav, S., Dharmela, G., Mistry, K. (2021). Fake review detection using machine learning techniques. International Journal of Emerging Technologies and Innovative Research, 8(4): 308-315. https://www.jetir.org/papers/JETIR2104042.pdf.

[3] Le, H., Kim, B. (2020). Detection of fake reviews on social media using machine learning algorithms. Issues in Information Systems, 21(1): 185-194. https://doi.org/10.48009/1_iis_2020_185-194

[4] Mohawesh, R., Xu, S., Springer, M., Al-Hawawreh, M., Maqsood, S. (2021). Fake or genuine? Contextualised text representation for fake review detection. arXiv preprint arXiv:2112.14343. https://doi.org/10.48550/arXiv.2112.14343

[5] Mohawesh, R. (2022). Machine learning approaches for fake online reviews detection. Doctoral dissertation, University of Tasmania. https://doi.org/10.25959/23246411

[6] Elmogy, A.M., Tariq, U., Ammar, M., Ibrahim, A. (2021). Fake reviews detection using supervised machine learning. International Journal of Advanced Computer Science and Applications, 12(1): 601-606. https://doi.org/10.14569/IJACSA.2021.0120169

[7] Bansode, M., Birajdar, A. (2021). Fake review prediction and review analysis. International Journal of Innovative Technology and Exploring Engineering, 10(7): 143-151. https://doi.org/10.35940/ijitee.G9042.0510721

[8] Pal, K., Poddar, S., Jayalakshmi, S.L., Choudhury, M., Saif Ahmed, S.K., Halder, S. (2022). Opinion mining-based fake review detection using deep learning technique. In 4th International Conference on Data Science, Machine Learning and Applications, Hyderabad, India, pp. 13-20. https://doi.org/10.1007/978-981-99-2058-7_2

[9] Monica, C., Nagarathna, N.J.S.C.S. (2020). Detection of fake tweets using sentiment analysis. SN Computer Science, 1(2): 89. https://doi.org/10.1007/s42979-020-0110-0

[10] Wang, J., Wu, C. (2020). Camouflage is NOT easy: Uncovering adversarial fraudsters in large online app review platform. Measurement and Control, 53(9-10): 2137-2145. https://doi.org/10.1177/0020294020970213

[11] Mohawesh, R., Xu, S., Tran, S.N., Ollington, R., Springer, M., Jararweh, Y., Maqsood, S. (2021). Fake reviews detection: A survey. IEEE Access, 9: 65771-65802. https://doi.org/10.1109/ACCESS.2021.3075573

[12] Mattson, C., Bushardt, R.L., Artino Jr, A.R. (2021). When a measure becomes a target, it ceases to be a good measure. Journal of Graduate Medical Education, 13(1): 2-5. https://doi.org/10.4300/JGME-D-20-01492.1

[13] Mewada, A., Dewang, R.K. (2022). Research on false review detection methods: A state-of-the-art review. Journal of King Saud University-Computer and Information Sciences, 34(9): 7530-7546. https://doi.org/10.1016/j.jksuci.2021.07.021

[14] Gupta, R., Jindal, V., Kashyap, I. (2024). Recent state-of-the-art of fake review detection: A comprehensive review. The Knowledge Engineering Review, 39: e8. https://doi.org/10.1017/S0269888924000067

[15] Alsaad, M.M.B., Joshi, H. (2024). Use of supervised machine learning classifiers for online fake review detection. Journal of Applied Optics, 49-70. https://appliedopticsjournal.net/index.php/JAO/article/view/86.

[16] Afifah, K., Yulita, I.N., Sarathan, I. (2021). Sentiment analysis on telemedicine app reviews using XGBoost classifier. In 2021 International Conference on Artificial Intelligence and Big Data Analytics, Bandung, Indonesia, pp. 22-27.

https://doi.org/10.1109/ICAIBDA53487.2021.9689762

[17] Asaad, W.H., Allami, R., Ali, Y.H. (2023). Fake review detection using machine learning. Revue d'Intelligence Artificielle, 37(5): 1159-1166. https://doi.org/10.18280/ria.370507

[18] Taşağal, K., Uçar, Ö. (2018). Detection of fake user reviews with deep learning. International Journal of Research in Engineering and Applied Sciences (IJREAS), 8(12): 8-15. https://d1wqtxts1xzle7.cloudfront.net/58224785/2_Dec-EAS-6215-libre.pdf.

[19] Kalbhor, S., Goyal, D., Sankhla, K. (2025). BERTConvNet: A transformer-based framework for aspect-based sentiment analysis and fake review detection on self-created YouTube review dataset. Ingenierie des Systemes d'Information, 30(6): 1639-1651. https://doi.org/10.18280/isi.300622

[20] Chenoori, R.K., Kavuri, R. (2022). GrFrauder: A novel unsupervised clustering algorithm for identification group spam reviewers. Ingénierie des Systèmes d'Information, 27(6): 1019-1027. https://doi.org/10.18280/isi.270619

[21] Raza, S., Paulen-Patterson, D., Ding, C. (2025). Fake news detection: Comparative evaluation of BERT-like models and large language models with generative AI-annotated data. Knowledge and Information Systems, 67(4): 3267-3292. https://doi.org/10.1007/s10115-024-02321-1

[22] Geetha, S., Elakiya, E., Kanmani, R.S., Das, M.K. (2025). High performance fake review detection using pretrained DeBERTa optimized with Monarch Butterfly paradigm. Scientific Reports, 15(1): 7445. https://doi.org/10.1038/s41598-025-89453-8

[23] Alshehri, A.H. (2024). An online fake review detection approach using famous machine learning algorithms. Computers, Materials and Continua, 78(2): 2767-2786. https://doi.org/10.32604/cmc.2023.046838

[24] Amulya, K., Swathi, S.B., Kamakshi, P., Bhavani, Y. (2022). Sentiment analysis on IMDB movie reviews using machine learning and deep learning algorithms. In 2022 4th International Conference on Smart Systems and Inventive Technology, Tirunelveli, India, pp. 814-819. https://doi.org/10.1109/ICSSIT53264.2022.9716550

[25] Kadhim, A.I. (2018). An evaluation of preprocessing techniques for text classification. International Journal of Computer Science and Information Security, 16(6): 22-32.

[26] Plisson, J., Lavrac, N., Mladenic, D. (2004). A rule based approach to word lemmatization. Proceedings of IS, 3: 83-86. https://aile3.ijs.si/dunja/SiKDD2004/Papers/Pillson-Lematization.pdf.

[27] Sıngh, S., Kumar, K., Kumar, B. (2024). Analysis of feature extraction techniques for sentiment analysis of tweets. Turkish Journal of Engineering, 8(4): 741-753. https://doi.org/10.31127/tuje.1477502

[28] Thakkar, A., Chaudhari, K. (2020). Predicting stock trend using an integrated term frequency–Inverse document frequency-based feature weight matrix with neural networks. Applied Soft Computing, 96: 106684. https://doi.org/10.1016/j.asoc.2020.106684

[29] Salminen, J., Kandpal, C., Kamel, A.M., Jung, S.G., Jansen, B.J. (2022). Creating and detecting fake reviews of online products. Journal of Retailing and Consumer Services, 64: 102771. https://doi.org/10.1016/j.jretconser.2021.102771