



ISSN: 2723-9535

Available online at [www.HighTechJournal.org](http://www.HighTechJournal.org)

# HighTech and Innovation Journal

Vol. 4, No. 2, June, 2023



## A Comparative Study of Sentiment Analysis Methods for Detecting Fake Reviews in E-Commerce

Maneerat Puttarattanamanee<sup>1</sup>, Laor Boongasame<sup>2, 3</sup>, Karanrat Thammarak<sup>4\*</sup>

<sup>1</sup> KMITL-Digital Analytics and Intelligence Center, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand.

<sup>2</sup> Department of Mathematics, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand.

<sup>3</sup> Business Innovation and Investment Laboratory: B2I-Lab, School of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand.

<sup>4</sup> Department of Computer Engineering and Electronics, School of Engineering and Technology, Walailak University, Nakhon Si Thammarat 80160, Thailand.

Received 05 February 2023; Revised 27 April 2023; Accepted 08 May 2023; Published 01 June 2023

### Abstract

The popularity of the e-commerce system has increased, especially under the COVID scenario. Consumer product reviews from the past have had a significant impact on influencing consumers' purchasing decisions. Fake reviews—those written by humans and computers that engage in dishonest behavior—are consequently generated to increase product sales. The fake reviews hurt consumers and are dishonest. The goal of this research is to examine and evaluate the performance of various methods for identifying fake reviews. The well-known and widely-used Amazon Review Data (2018) dataset was used for this research. The first 10 product categories on Amazon.com with favorable feedback will be provided in the data section. After that, perform fundamental data preparation procedures such as special character trimming, bag of words, TF-IDF, etc. The models are trained to create a dataset for detecting fake reviews. This research compares the performance of four different models: GPT-2, NBSVM, BiLSTM, and RoBERTa. The hyperparameters of the models are also tuned to find the optimal values. The research concludes that the RoBERTa model performs the best overall, with an accuracy of 97%. GPT-2 has an overall accuracy of 82%, NBSVM has an overall accuracy of 95%, and BiLSTM has an overall accuracy of 92%. The research also calculates the Area Under the Curve (AUC) for each model and finds that RoBERTa has an AUC of 0.9976, NBSVM has an AUC of 0.9888, BiLSTM has an AUC of 0.9753, and GPT-2 has an AUC of 0.9226. It can be observed that the RoBERTa model has the highest AUC value, which is close to 1. Therefore, it can be concluded that this model provides the most accurate prediction for detecting fake reviews, which is the main focus of this research.

**Keywords:** Fake Reviews Detection; GPT-2; NBSVM; BiLSTM; RoBERTa.

## 1. Introduction

E-commerce systems have continued to gain popularity [1]. The worse the COVID-19 situation, the more clearly its popularity has increased. This is because people cannot travel outside their homes. As for companies, they benefit from trading on the e-commerce system; that is, they can reduce operating costs, such as rental fees for trading establishments. It is also possible to easily expand their business to allow foreigners to trade with them. As for customers, they benefit from trading on the e-commerce system, namely by saving time traveling. Customers can compare products quickly and easily.

\* Corresponding author: [kanchan.th@wu.ac.th](mailto:kanchan.th@wu.ac.th)

 <http://dx.doi.org/10.28991/HIJ-2023-04-02-08>

➤ This is an open access article under the CC-BY license (<https://creativecommons.org/licenses/by/4.0/>).

© Authors retain all copyrights.

There are many factors that influence a customer's decision to purchase products or services on the e-commerce system, such as price, quality, reliability, service, and payment security. There is also another factor that is very important nowadays: reviews and opinions from previous customers or consumers. Such customer reviews and opinions are seen as more neutral and credible as compared to the promotional content created by the company. If the reviews are positive, customers will have confidence until they finally decide to buy the product [2].

On social media, however, there are both true and fake reviews and opinions from prior clients or consumers. Fake customer reviews and opinions may be created by a business to boost sales of its goods and services, decrease sales of those of its competitors' goods and services, or even by customers who have never used the product in question. Fake customer reviews and opinions will harm the company that sells the goods and services [3], resulting in the deterioration of the company's reputation and the loss of sales. As for the customers, they will receive a product that does not meet their needs, wasting money and losing their feelings.

Many techniques have been attempted to detect fake customer reviews and opinions, such as sentiment analysis [4], reviewer profile analysis [5], language and text analysis [6], review duration and integrating reviews [7], machine learning algorithms [8], and image analysis [9]. One of the most popular techniques is the sentiment analysis method. This includes checking the original content in order to distinguish emotions or hidden feelings [10]. This helps detect fake reviews. This often reveals inconsistencies between the textual content and the emotions or ratings expressed [11]. There are several techniques used in sentiment analysis, such as Generative Pretrained Transformer 2 (GPT-2) [12], Naive Bayes Support Vector Machine (NBSVM) [13], Robustly Optimized BERT Pretrained Approach (RoBERTa) [14], and Bi-Directional Long Short-Term Memory Model (BiLSTM) [15].

However, there has been no research that has analyzed, compared, and evaluated the effectiveness of various techniques used in sentiment analysis. In this study, a comparative analysis and evaluation of the effectiveness of various techniques used in sentiment analysis were performed. It helps to gain a better understanding of those techniques, including the pros and cons of each technique under different circumstances, using data preparation techniques such as word wrapping, special letter wrapping, and word splitting before comparative evaluation.

The subsequent sections of this paper are organized in the following manner: Sections 1 and 2 provide an introduction to the subject matter and present relevant literature studies. Section 3 provides a comprehensive description of the experimental setup and technique employed in the study. Section 4 presents a comprehensive analysis of the efficacy of the recommended methodologies. Section 5 of this paper presents an analysis of the study's findings and compares them with those of other relevant studies. In conclusion, Section 6 provides a comprehensive overview of the primary outcomes derived from the study.

## 2. Background and Related works

### 2.1. Background

#### 2.1.1. Sentiment Analysis

Sentiment analysis [16, 17] entails the examination of emotions and sentiments, both positive and negative, conveyed through various forms of communication, such as movie reviews, restaurant reviews, online product feedback, and more. It is an effective method for evaluating written or spoken language to determine whether a thought expression is positive, negative, neutral, or potentially designed to deceive or persuade for various purposes. Sentiment analysis has significant value in the business world because it enables companies and brands to gauge consumer sentiments regarding their products or services. This invaluable insight contributes to the development and enhancement of offerings, resulting in increased consumer satisfaction. Sentiment analysis is a subset of Natural Language Processing (NLP), a technique used to classify text based on the emotional content it conveys. E-commerce platforms commonly use sentiment analysis to gauge consumer opinion about products and determine their preferences and inclinations. Additionally, to conventional sentiment analysis, many other techniques are used to assess emotions and attitudes within the text [18]. These approaches include emotion detection, which identifies specific emotions like happiness or anger, and aspect-based sentiment analysis, which dissects text into distinct attributes for separate analysis. Deep learning models [8] such as RNNs and transformers are used for understanding complex nuanced contexts. Lexicon-based analysis [19] uses predefined sentiment scores.

Sentiment analysis is the practice of interpreting and removing content from text that expresses emotions or opinions using structured methods. This procedure comprises several stages as shown in Figure 1.

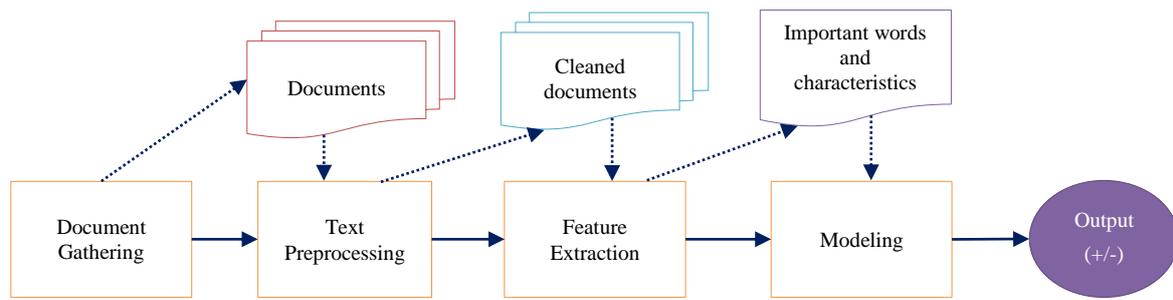


Figure 1. Sentiment analysis process

Initially, pertinent text data is gathered from a variety of sources, including social media, consumer reviews, polls, and other text-based sources. After that, the data are cleaned and prepared for analysis through a procedure named text pre-processing. This stage includes tasks such as removing punctuation, changing text to lowercase, addressing special characters, and tokenizing the text by separating it into distinct words or phrases.

The next step is featuring extraction, which involves finding important textual characteristics or words that are indicative of sentiment. Methods such as bag-of-words (BoW) [20] and term frequency-inverse document frequency (TF-IDF) [21] are two examples of techniques that could be used to capture the data. The classification of emotions is the fundamental building block of sentiment analysis. In this stage, a bespoke machine learning model is trained to analyze the text into several sentiment categories, such as positive, negative, or neutral. After training, the sentiment analysis model labels and scores each text element in the dataset. Finally, sentiment analysis informs changes to the input, model, or methodology to increase accuracy and relevance for future assessments.

### 2.1.2. Tokenization

Tokenization is the process of preparing data by dividing text or sentences into words. Put it in the form of a single word or group of words [8]. It then clusters different words and counts the number of words of interest in the dataset. This is named a token and is often used for other linguistic analyses. In tokenization, words or phrases have intrinsic meaning but when joined with others, new meanings can be created. In addition, the sequence patterns of tokens can be analyzed using the N-gram method. Commonly used in such analyses are 1-gram, or mono-gram tokens, for analyzing individual words, and 2-gram, or bi-gram tokens, for two adjacent words.

### 2.1.3. Bag of Words

The Bag of Words (BoW) technique [20] is a text feature extraction method that employs the concept of One-Hot Encoding to represent the information contained in a dataset's words. In this approach, each word in the dataset is associated with a unique token. When a word appears in the dataset, it is represented by a token with a value of 1. Conversely, when a word does not appear in the dataset, its corresponding token is set to 0 during encoding, effectively indicating its absence. This encoding method is also applied in encryption to transform text data for various purposes. As shown in Figure 2.

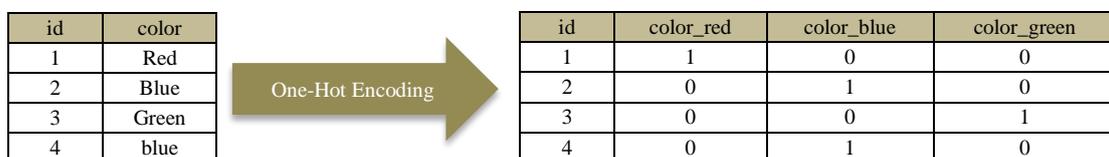


Figure 2. Example of encoding using One-Hot Encoding technique

The Bag of Words (BoW) method transforms text into a format that is easily processed by computers and serves as the foundation for various algorithms. BoW involves tokenization, which is the process of breaking text into individual units or tokens. These tokens are then statistically analyzed to gather information about the words present in the text. This process results in the creation of a vocabulary based on the dataset being used. Typically, words are ranked by frequency, with the most frequently occurring tokens given the highest priority as the primary keys for representation in further analysis.

BoW offers versatility in its application. It can be employed for tasks such as data analysis, where it helps in constructing a vocabulary or a collection of words relevant to the dataset. However, BoW has limitations in counting occurrences. Specifically, it can count the frequency of words but cannot determine how many documents contain a particular word. This limitation becomes problematic when certain words appear frequently or rarely in a single document. In such cases, the data analytic precision may be compromised.

### 2.1.4. Term Frequency – Invert Document Frequency

The TF-IDF [21] method is a statistical approach used for determining the relevance of words in a document. The calculation is derived by multiplying the term frequency variable with the inverse document frequency variable. The phrase "term frequency" (TF) denotes the frequency at which a specific word occurs in a document. As an illustration, the definite article "the" may appear ten times in a given document, whereas the noun "dog" may be present twice. The inverse document frequency (IDF) is a metric that quantifies the rarity of a word in a given corpus of documents. The word "the" is frequently used and has a relatively low inverse document frequency (IDF), indicating its high occurrence across various texts. Conversely, the term "dog" is less commonly used and has a larger IDF, suggesting its lower prevalence in different contexts. The calculation of the TF-IDF value for a word involves the multiplication of its term frequency by its inverse document frequency. As an illustration, the term frequency-inverse document frequency (TF-IDF) value of the word "the" in a document can be computed as follows:

$$TF-IDF = (term\ frequency) \times (inverse\ document\ frequency) = 10 \times (1 / \log(number\ of\ documents)) \tag{1}$$

The weighting of a word in a document can be determined by calculating its TF-IDF score. Words with a high term frequency-inverse document frequency (TF-IDF) are more likely to be evaluated significant compared to words with low TF-IDF values.

### 2.1.5. Word Embedding

Word embedding is an indispensable natural language processing technique that converts textual data such as words, sentences, and documents into numerical feature vectors [21]. These vectors depict the semantic meaning and relationships of words, which facilitates mathematical operations and calculations in language-based tasks. Typically, the process begins with the encoding of words using methods such as One-Hot Encoding or advanced techniques such as Word2Vec and GloVe, yielding feature vectors the dimensions of which are determined by the vocabulary size and embedding method. These vectors facilitate the measurement of word similarity across various linguistic contexts, thereby making word embedding a valuable resource for numerous language-related applications.

### 2.1.6. Bidirectional Long Short-Term Memory

A Bidirectional Long Short-Term Memory (BiLSTM) [15] is a specialized type of recurrent neural network (RNN) architecture designed to enhance the capabilities of traditional LSTMs (Long Short-Term Memory networks). What sets BiLSTMs apart is their ability to process input sequences in two directions: both forward and backward. During the forward pass, the input sequence is analyzed from beginning to end, similar to a regular LSTM, capturing information from past time steps. Simultaneously, a second LSTM processes the input sequence in reverse, from the end to the beginning, capturing information from future time steps. Afterward, the outputs of these two LSTMs are typically combined or concatenated to provide a holistic representation of the sequence. Bi-LSTM cells feature numerous layers for each iteration T, including an input layer  $X_t$ , an output layer  $h_t$  and a hidden layer  $h_{t-1}$ . Every cell shares some states with other cells during training or parameter updates as shown in Figure 3.

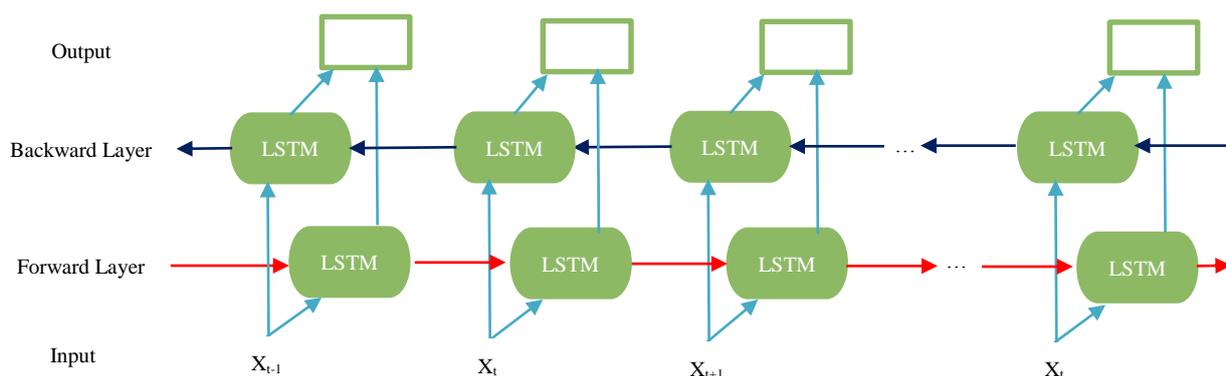


Figure 3. Bidirectional Long Short-Term Memory Architecture [22]

The Bi-LSTM architecture has demonstrated promising performance in a variety of NLP tasks, such as sentiment analysis, named entity recognition, and machine translation. The mathematical model of a Bi-LSTM consists of the forward and backward LSTM equations as well as the concatenation of their outputs. The backward LSTM equations are similar to the forward LSTM equations, but with distinct weight matrices and bias terms. The final result of the Bi-LSTM model is obtained by concatenating the outputs of the forward and backward LSTMs at each time phase. During the training process, the model parameters (weights and biases) are learned using techniques such as backpropagation through time (BPTT) and gradient descent. Its capacity for capturing information from both past and future contexts makes it suitable for tasks where comprehending the input sequence requires knowledge of the surrounding context.

### 2.1.7. Generative Pre-trained Transformer 2

GPT-2 [12], also known as Generative Pre-trained Transformer 2, is a large language model developed by OpenAI. This model is a part of the fundamental series of GPT models. The training data utilized by GPT-2 encompassed the BookCorpus dataset, which encompassed a vast collection of more than 7,000 unpublished works of fiction spanning several genres. Additionally, the training data included a dataset encompassing a staggering 8 million web pages. The initial partial release of the model took place in February 2019, and it was then followed by the full release of the 1.5-billion-parameter version on November 5, 2019. GPT-2, similar to its preceding and subsequent models, is constructed based on a generative pre-trained transformer architecture. The architectural design of this system incorporates a deep neural network, more specifically a transformer model, which leverages attention mechanisms as opposed to traditional recurrence- and convolution-based methodologies. The attention mechanism allows the model to focus on selected segments of input text that it deems most relevant. GPT-2 offers various model size options (124 M, 774 M, etc.) that are distinguished primarily by the number of transformer decoders layered in the model; however, all models have the same core component, which consists of a series of transformer decoders, each of which has an identical architectural structure, as shown in Figure 4.

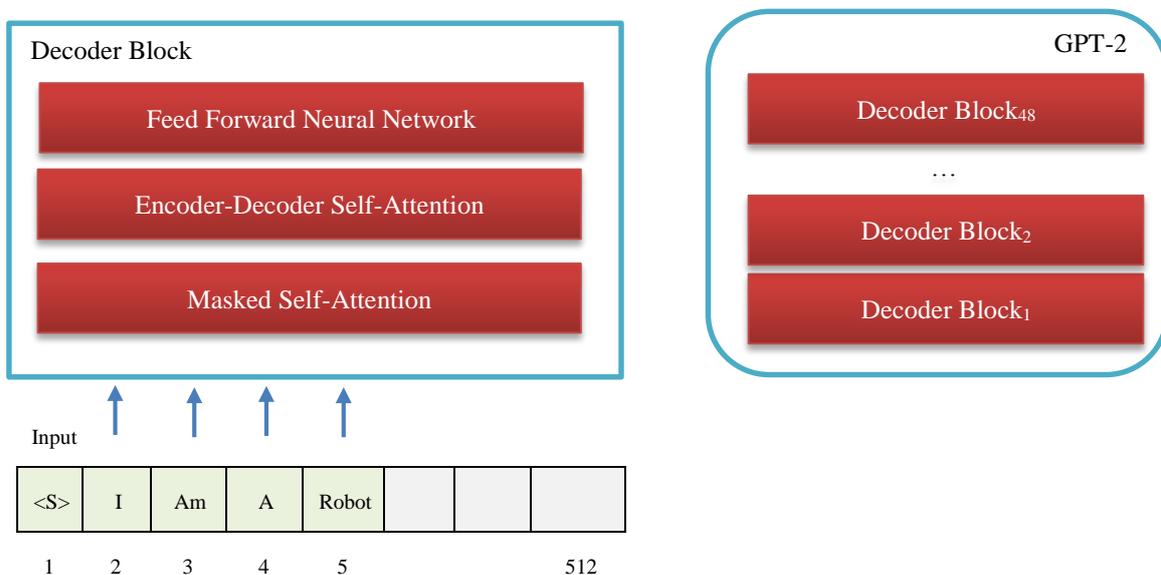


Figure 4. GPT-2 architecture [23]

### 2.1.8. Robustly Optimized BERT Approach

RoBERTa (Robustly Optimized BERT Approach) [14] is a transformer-based language model that employs self-attention to evaluate input sequences and construct contextualized representations of sentence terms. RoBERTa was developed by University of Washington researchers using a training dataset comprised of > 10 times the text in BERT's training dataset. RoBERTa's training technique was also significantly more effective. In addition, when training the model, RoBERTa employs a dynamic masking strategy. This technique enables the model to acquire more trustworthy and generalizable word representations. RoBERTa was trained on multiple text datasets, such as the English Wikipedia, the BOOK CORPUS, CC-NEWS, and OPENWEBTEXT. STORIES was employed as well. The cumulative size of these diverse textual datasets was in the tens of terabytes. By providing several tens of terabytes of data, they supplied the model with a solid linguistic foundation.

The BERT [24] architecture was modified to create RoBERTa, which included several substantial modifications to enhance performance on natural language comprehension tasks. RoBERTa is a variant of the BERT architecture that underwent enhancement. RoBERTa eliminated the Next Sentence Prediction (NSP) objective that was part of BERT's training procedure. This modification maintained or marginally enhanced subsequent task performance, indicating that NSP is not a crucial element of the training. In the second phase of RoBERTa's training, increased quantities and longer sequences were utilized. The training for RoBERTa consisted of 125,000 steps with 2,000 sequences and 31,000 steps with 8,000 sequences, resulting in increased task complexity and precision. BERT was trained using 256 sequences per cohort and one million steps. Additionally, this strategy simplified and enhanced parallelization., RoBERTa utilized dynamic masking for data preparation in contrast to BERT, which utilized a static masker. Over forty epochs, the same

data was replicated and disguised ten times using a variety of masking techniques. This was done for each era to have a distinct selection of masking patterns. Consequently, RoBERTa's performance was significantly enhanced by this dynamic training data diversification technique.

### 2.1.9. Naive Bayes Support Vector Machine

A Naive Bayes Support Vector Machine, or NBSVM [13], is a hybrid approach to machine learning that combines Naive Bayes and Support Vector Machines that can be advantageous in certain situations. Naive Bayes, which is renowned for its simplicity and efficiency, can be utilized for initial classification or feature selection. This method is especially useful for text classification problems and situations where the assumption of feature independence remains reasonably true. Conversely, SVM, a robust algorithm for discovering complex patterns, can be used to extract more intricate representations or patterns from the data. By integrating the strengths of Naive Bayes and SVM, this hybrid method seeks to improve classification performance overall. Typically, the process involves feature engineering, where pertinent features are extracted from the dataset or created. Then, a Naive Bayes classifier is trained, and SVM is used to further process or refine the data, perhaps with a different subset of features. The predictions or representations from both models are then typically combined using stacking or ensemble methods. This combination's efficacy is highly dependent on the specific dataset and problem at hand. It is essential to perform a comprehensive data analysis, experiment with various approaches, and use cross-validation to determine whether the hybrid approach offers advantages over other ensemble methods or using each algorithm individually.

## 2.2. Related Works

Currently, there exist various approaches to analyzing emotions and sentiments from textual data. There are various approaches to analyzing emotions and sentiments from textual data, including rule-based Sentiment Analysis [18], lexicon-based sentiment analysis [19], as well as machine learning and deep learning models [8]. Currently, there are various approaches to analyzing emotions and sentiments from textual data. These include rule-based Sentiment Analysis [18], lexicon-based sentiment analysis [19], as well as machine learning and deep learning models [8]. This study employed machine learning and deep learning models for sentiment analysis and identification of fake reviews. The investigation's primary objective was to develop algorithms capable of effectively discerning fake reviews. Additional information regarding the subject matter can be found in Table 1.

As Table 1 shows, numerous empirical investigations were conducted, yielding a range of modeling methodologies that efficiently and accurately forecasted the detection of fake reviews across diverse datasets including e-commerce platforms, hotels, and restaurants. Furthermore, the researchers conducted analyses on standardized review datasets obtained from various platforms, including Yelp, Amazon, Kaggle, Github, Google Play Store, AppStore, and TripAdvisor.

Several researches incorporate sentiment analysis and feature selection techniques in order to improve the performance of models. Vidanagama et al. [25] conducted a study with the objective of identifying fake reviews in the field of e-commerce. Their approach involved the utilization of ontology-based techniques and linguistic features. The study conducted a comparative examination of predictive performance by evaluating the association between significant features. The findings revealed that integrating linguistic features with word classification and sentiment analysis yielded a prediction accuracy of 88.98%. Rathore et al. [29] used a semi-supervised methodology to identify groups of fake reviewers through the utilization of a semi-supervised methodology. The investigation employed a dataset sourced from the Google Play Store. The study evaluated the respective performance of the SVM, K Nearest Neighbor (kNN), Random Forest (RF), and J48 algorithms.

Notably, their performance surpassed that of comparable alternatives. In a study using a dataset from Yelp, which included 20 hotel reviews, Tufail et al. [30] examined the effects of detecting fake reviews. The prediction models employed for performance comparison in this work included SVM, logistic regression (LR), and kNN. The research revealed that the highest level of practical efficiency was attained by integrating all three algorithms, leading to a notable accuracy rate of 95%. Qayyum et al. [26] produced the FRD-LSTM, a deep learning-based method for detecting fake reviews, by using the DCWR algorithm to compute deep features and PCA to reduce the feature space. Training the Bi-LSTM helps to detect fake reviews. The results indicate that this method has a recall rate of 97.24% precision of 96.0%, an F1-Score of 96.6%, and accuracy of 97.21%. In addition, recent research has also focused on large language models such as GPT-2 [27], RoBERT [27], and BERT [31], which exhibit a remarkable level of accuracy and remain of great interest for the foreseeable future.

Table 1. Related works

Article	Objective	Technique	Dataset	Best Performance
Vidanagama et al. [25]	The aim of this study is to identify fake reviews in the realm of electronic commerce by the utilization of ontology-based linguistic sentiment analysis.	1. Text Preprocessing 2. Linguistic Feature 3. POS Feature Extraction 4. Ontology Management	Kaggle: Amazon Unlocked Mobile Reviews	Ontology Management Accuracy = 88.98%
Qayyum et al. [26]	This study introduced FRD-LSTM, a deep learning-based fake review detection method. DCWR computes deep features after preprocessing. PCA reduces feature space. Fake reviews are identified by training the Bi-LSTM classifier with computed attributes.	1. DCWR algorithm 2. Bi-LSTM	Amazon	Bi-LSTM + DCWR Accuracy = 97.21%
Salminen et al. [27]	This study attempts to use the ULMFIT and GPT-2 models to generate and detect fake reviews within a dataset obtained from the Amazon e-commerce platform.	1. OpenAI 2. NBSVM 3. RoBERTa	Amazon e-commerce dataset	RoBERTa Accuracy = 96.64%
Yao et al. [28]	To compare machine learning methods for detecting fake reviews and improve model accuracy by balancing data using SMOTE and RUS concepts.	1. RF 2. LSTM 3. SVM 4. CNN	Amazon Mechanical Turk	RF + Feature ranking 90% F-Score
Rathore et al. [29]	Utilize a hierarchical methodology to detect clusters of fake reviewers by implementing the DeepWalk technique.	1. SVM 2. kNN 3. RF 4. J48	Google Play Store	kNN Accuracy = 87.00%
Tufail et al. [30]	A Bigram model, along with the SVM, kNN, and SKL algorithms, can be used to detect fake reviews. The study would look at how many pronouns, verbs, and feelings were used.	1. SVM 2. kNN 3. SKL	Yelp: hotel reviews	SKL Accuracy = 95.00%
Mir et al. [31]	This paper proposes using the BERT model to extract words from reviews and the SVM model to incorporate the extracted words.	1. BERT 2. SVM	hotel, doctor, restaurant, amazon	BERT+SVM Accuracy 87.81%
Sohan et al. [32]	To detect fake product reviews using semi-supervised machine learning and feature engineering to extract varied reviewer behaviors.	1. RF	Kaggle: food review dataset	RF 97.7 % F-Score
Alsubari et al. [33]	To develop of integrated CNN-LSTM Model for Identifying fake reviews in E-Commerce using multidomain datasets.	1. CNN-LSTM	Hotels, Restaurants, Yelp, and Amazon	CNN-LSTM Accuracy = 89.00%
Shetgaonkar. et al. [34]	To identify fake reviews using sentiment analysis and deep learning neural networks like the GRU, Bi-LSTM, and LSTM.	1. GRU 2. Bi-LSTM 3. LSTM	Amazon e-commerce dataset	GRU Accuracy = 92.00%
Elmoqy et al. [35]	To examine different machine learning techniques in detecting false reviews and enhance the model's accuracy by applying SMOTE and RUS principles to balance the data.	1. RF 2. LSTM 3. SVM 4. CNN	Amazon Mechanical Turk	kNN Accuracy = 82.40%
Nagi Alsubari et al. [36]	utilizing n-grams and reviewer sentiment ratings to detect fake reviews on e-commerce platforms. It extracted features using data preprocessing and TF-IDF.	1. NB 2. SVM 3. AB 4. RF	Trip Advisor: hotel review dataset	RF Accuracy = 95.00%

**Abbreviation:**

BiLSTM= Bidirectional Long Short-Term Memory/LSTM= Long Short-Term Memory/NB= Naive Bayes/SVM= Support Vector Machines/NBSVM= Naive Bayes (NB) and Support Vector Machines/DT= Decision Tree/GRU= Gated Recurrent Unit/CNN= Convolutional Neural Network/MLP= Multilayer Perceptron/GNB= Gaussian Naive Bayes/ RoBERTa= A Robustly Optimized BERT Pretraining Approach. OpenAI or GPE-2= Generative Pre-trained Transformer 2/LR= Logistic Regression/ SKL= K-Nearest Neighbor, and logistic regression /AB= AdaBoost).

### 3. Methods

The main objective of this study was to compare various models and provide methodologies for detecting fake reviews in e-commerce platforms by using machine-learning and deep-learning techniques. As depicted in Figure 5, the study's conceptual framework consists of two primary phases: data preparation for computation and model training for detection.

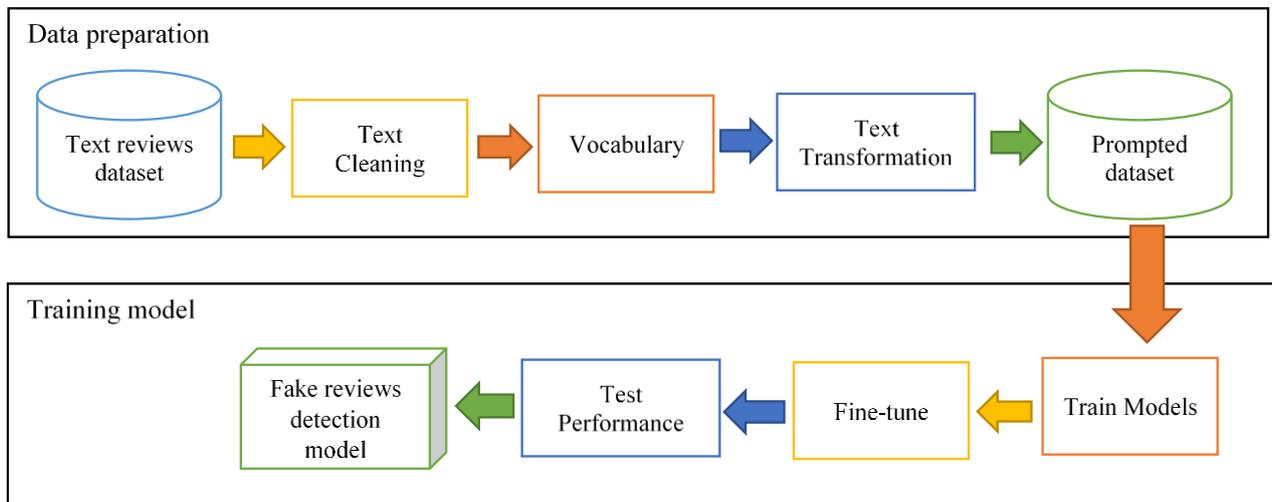


Figure 5. Conceptual framework

#### 3.1. Data Preparation Process

Data preprocessing is an essential part of this study that aims to enhance the efficiency of data analysis and reduce the complexity of the model. In the domain of sentiment analysis, the process of data preparation encompasses multiple stages. These stages include the cleansing of data by eliminating special characters or symbols, transforming all letters to lowercase, segmenting sentences into individual words, removing sentence-ending words, and discerning various word categories, including nouns and verbs. This study employed an English-based methodology for data preparation. This process consists of three stages, as shown in Figure 5.

In the initial stage, the textual dataset must be transformed into encrypted numerical representations that are computationally feasible, therefore, this step entails text cleaning, which involves eliminating errors and inconsistencies in the text dataset. It also involves correcting misspelled words and establishing consistent formatting between uppercase and lowercase letters. Additionally, the process eliminates symbols and unnecessary elements from the collection. The subsequent stage establishes a lexicon for analysis. This process encompasses collecting words from phrases and using varied terminology to expand one's vocabulary. Quantifying the frequency of the terms that most significantly influence the dataset, can determine the scope of the term. This process helps to exclude extraneous data during the analysis phase. The final stage is text transformation which includes converting words into a format compatible with computer systems, to allow processing.

#### 3.2. Dataset

The dataset included in this study was curated by Ni & McAuley [37]. It comprises product reviews sourced from Amazon.com. The study centered on the ten product categories with the most reviews in 2018 including Beauty (5,269), Automotive (1,711,519), Gift Cards (2,972), Magazine Subscriptions (2,375), Fashion (3,176), CDs & Vinyl (1,443,755), Groceries & Gourmet (1,143,860), Musical Instruments (231,392), Appliances (2,277), Cell Phones & Accessories (1,128,437), Industry and Science (77,071), Office products (800,357), Handicrafts and Sewing (494,485), Digital Music (169,781), Luxury and Beautiful Products (34,278), and Yard, Lawn, and Garden (798,415).

The data was separated into a variety of categories, such as Toys and Games, Kindle Store, Pet Supplies, Sports and Outdoors, Tools and Home Improvement, and Books, Clothing, Shoes, and Jewelry. Other categories were Electronics, Home and Kitchen, Movies and TV, and the Kindle Store. In total, 40,432 samples were divided between the two categories after being analyzed. The first batch consisted of fake reviews that were generated by a machine and were labeled as computer-generated (CG). The second group included genuine evaluations written by individuals in their own words. These reviews were designated as original reviews (OR). The contents of Table 2 are presented in two sets, each of which consists of 20,216 rows and four columns.

**Table 2. Examples of dataset used in study**

Category	Rating	Label	Text
Home_and_Kitchen_5	1	CG	Missing information on how to use it, but it is a great product for the price! I
Clothing_Shoes_and_Jewelry_5	3	OR	Fits well and easy to convert with semi formal wear
Electronics_5	5	CG	Easy to install even without a power supply. Easy to use. Great product! Works
Pet_Supplies_5	2	CG	The holes are too big the fence receiver slipped out
Toys_and_Games_5	3	OR	Smaller than we thought. Granddaughter enjoyed painting it.

### 3.3. Training the Model

After converting the dataset into a computer compatible format, training proceeded using deep-learning and machine-learning techniques. Four models, GPT-2, NBSVM, RoBERTa, and BiLSTM were utilized to assess the effectiveness of detecting false reviews. For effective training, the model environment and parameters must be appropriately established. The following steps outline the procedures required to achieve optimal results.

#### 3.3.1. GPT-2 Setup

This study employed GPT-2, a large language model designed for unsupervised multitask learning and supervised learning. To eliminate inconsistencies, the dataset was pre-processed and annotated appropriately. To begin, a Keras tokenization library was invoked to generate the Bag of Words (BoW), which represented the entire vocabulary, from the supplied text. Python's tokenizer function was used to determine the maximum length of a word in a phrase. To achieve uniformity in sentence length, a numerical representation was assigned to each word in a sentence, with zeros added to complete the phrase. Before feeding the data into the model, it was divided into two subsets: the training set, which consists of 80% of the data, and the test set, which consists of 20% of the data.

#### 3.3.2. NBSVM Setup

The NBSVM model is widely employed in the field of natural language processing (NLP) due to its exceptional robustness. The study utilized additional techniques, such as the removal of unusual characters from the review text and the implementation of a BoW approach to classify groups of words or text. The transformation of label features into target values ultimately resulted in the production of target features. The numerical value of 1 was allocated to OR (representing real reviews), whereas CG (representing fake reviews) was assigned a numerical value of 0. The dataset was then divided into two sets: the training set, which comprised 80% of the data, and the test set, which comprised the remaining 20%.

#### 3.3.3. RoBERTa Setup

The initial value of the RoBERTa model was determined based on the information presented in Table 3 during the training phase. This function initiated the training process with a 0.10 dropout rate. The model was fine-tuned by incorporating various hyperparameters to evaluate its performance. This study intended to identify the optimal model for detecting fake reviews. The Results section elaborates on the findings in complete detail. The dataset was divided into two subsets, with 80% of the data designated for training and 20% reserved for testing.

**Table 3. The initial hyperparameter values for the RoBERTa algorithm's firstround model**

Hyperparameter	Initial Value
Optimizer	Adam
Learning Rate	0.00001
Epochs	1
Batch Sizes	8

#### 3.3.4. BiLSTM Setup

The BiLSTM model was selected as the method for detecting fake evaluations in this study. Bidirectional learning was implemented by the model to produce precise results to reduce data loss and enhance performance analysis. The data underwent pre-processing by removing special characters and frequently occurring stop terms. Before being input into the model, the data was partitioned into a training set containing 80% of the total data and a testing set containing the remaining 20%. The model's parameters were then set, and it was trained with a 0.5 percent dropout rate. Table 4 contains additional parameter specifications. In the last step, the number of training cycles and dropout rate were modified for more in-depth analysis and to improve the model's overall performance.

**Table 4. The initial hyperparameter values for the BiLSTM a algorithm's firstround model**

Hyperparameter	Initial Value
Loss Function	Binary Cross Entropy
Optimizer	Adam
Learning Rate	0.01
Epochs	5
Batch Sizes	128

**3.3.5. Defining the BiLSTM and RoBERTa Models for Fine-tuning**

To establish the model, the chosen configuration is Sequential, and the Adam optimizer is employed to determine the arrangement of the layers, which will be sequentially organized as follows.

- The input layer receives the incoming data.
- The embedding layer encapsulates word representations.
- The bidirectional layer utilizes parameter 150, which was employed by the BiLSTM.
- The dimension size of the dense layer was set to 32, and Rectified Linear Unit (ReLU) was the activation function.
- Dropout layer: Dropout is a technique used to reduce the information in a system by randomly deactivating certain nodes.
- Dense layer: This layer decreases the dimensionality of the input by reducing it to a single dimension.
- The second dropout layer also randomly deactivates nodes.
- The batch normalization layer normalizes data to address the problem of overfitting.
- The activation function for the dense layer, which serves as the output layer, should be set to sigmoid. This activation function ensures that the output values fall within the range of 0 to 1.

The Bidirectional Layer mitigates in accordance with the model's definition serves the purpose of that may occur the process of weight adjustment. Furthermore, the batch normalization layer and particularly the activation function, ReLU, effectively addresses the vanishing gradient problem and overfitting. Incorporating the dropout layer additionally serves to reduce these concerns.

**4. Results and Discussion**

**4.1. Results**

This section presents the findings of evaluating the performance of the GPT2, BiLSTM, RoBERTa, and NBSVM, models including the hyperparameter tuning process. The results are reported in three sections: 1) the performance test, 2) the hyperparameter tuning process, and 3) the receiver operating characteristic (ROC) curves and calculating the area under the curve (AUC).

The first section applies four metrics: accuracy, precision, recall, and F1-score. The dataset used for evaluation was the Amazon Review Data (2018) standard dataset [37]. The performance test results are presented in Table 5.

**Table 5. The result of the performance test for each algorithms**

Model	Accuracy	Precision	Recall	F1-Score
GPT-2	0.82	0.83	0.82	0.82
NBSVM	0.95	0.95	0.95	0.95
BiLSTM	0.92	0.93	0.93	0.92
<b>RoBERTa</b>	<b>0.97</b>	<b>0.97</b>	<b>0.97</b>	<b>0.97</b>

With the GPT-2 algorithm, the model's overall performance had an accuracy of 82.00%, with a precision of 76.00% for CGs and an accuracy of 90.00% for ORs. The recall rate for CGs was 92.00% and 72.00% for ORs. Additionally, the F1-score for CGs was 84.00%, and the F1-score for ORs was 80.00%. The NBSVM algorithm had an accuracy of 95.0%, a CG accuracy of 96.00%, and an OR precision of 94.00%. The recall rate for CGs was 93.00%, and the recall rate for ORs was 96.00%. The F1-score for CGs and ORs was 95.0%.

The BiLSTM algorithm exhibited a level of accuracy of 92.00% and precision of 91.00% in identifying CGs and precision of 93.00% in identifying ORs. The recall rate for identifying CGs was found to be 93.00%, while the recall

rate for identifying ORs was 91.00%. The F1-score achieved for distinguishing between CGs and ORs was 92.00%. The accuracy was classified according to its type. The category with the highest accuracy, namely 91.911162%, was in the "Books" category. This accuracy rate surpassed that of the other categories, which had an average accuracy of 91.05348%.

The experimental investigation of identifying CGs with the RoBERTa algorithm yielded noteworthy findings. Specifically, the results presented in Table 5 demonstrated excellent overall accuracy of 97.00%. The detection rate for identifying CGs was found to be 97.00% regarding precision, while the identification rate for ORs was determined to be 98.00% in terms of accuracy. The recall rate for identifying CGs was found to be 98.00%, while the recall rate for identifying ORs was 97.00%. The F1-score achieved for the classification of CGs and ORs was 97.00%.

The second section presents the results of the hyperparameter tuning process. For the BiLSTM and RoBERTa models, this study also fine-tuned the model by tuning hyperparameters to find the optimal values. The RoBERTa model had an initial dropout value of 0.1, and the BiLSTM had an initial dropout value of 0.5. The related hyperparameters are presented in Table 6.

**Table 6. Initial Hyperparameter value setting of RoBERTa and BiLSTM model**

Model	Parameter	Fine-tuning	Loss Function	Accuracy
BiLSTM	Learning Rate	0.01	0.2579	92.3564
		0.001	0.3697	91.0597
	Epochs	5	0.2579	92.3564
		10	0.5927	91.0102
	Batch Size	128	0.2579	92.3564
		120	0.7806	90.8742
RoBERTa	Learning Rate	0.00001	0.1202	97.0276
		0.0001	0.6987	50.8822
	Epochs	1	0.1202	97.0276
		5	0.1175	96.3645
	Batch Size	8	0.1202	97.0276
		16	0.1196	94.7322

Tests were conducted on several hyperparameters to fine-tune the BiLSTM algorithm. The adjustment of the learning rate to 0.001 caused the model's accuracy to decline from 92.3564% to 91.0597%, and loss to increase from 0.2579 to 0.3697. Hence, the inferred ideal learning rate for the BiLSTM model is 0.01. The subsequent phase involved optimizing the number of epochs for training by manipulating the epochs parameter. After a series of tests involving different numbers of epochs, the model was found to perform best when trained for five epochs. Finally, batch sizes of 128 and 120 were compared to ascertain the number of iterations required for the model to learn the data prior to weight adjustment. The study revealed that the BiLSTM model achieved higher accuracy with a batch size of 128, rather than 120.

Therefore, in assessing the Amazon Reviews Dataset (2018) using the BiLSTM model, the best hyperparameters for obtaining the most accurate results were: learning rate = 0.01, epochs = 5, and batch size = 128. The resulting accuracy achieved with these hyperparameters was 92.35%.

In evaluating the RoBERTa model, optimal performance was attained when trained with a learning rate of 0.00001 compared with a 0.0001 learning rate. The subsequent step involved modifying the batch size, which was initially set at 8, to a new value of 16. The result revealed that altering the batch size to 16 resulted in its efficiency declining from 97.0276 to 94.7322. Hence, 8 was determined to be the best batch size for the RoBERTa model. Subsequently, the epoch value was adjusted to 5 to find the most favorable number of training cycles for the model. The data presented in the table indicates that there was a drop in model efficiency when the value of epochs was changed from 1 to 5.

Based on these findings, the optimal hyperparameters for the RoBERTa model applied to the Amazon Reviews Dataset (2018) are: a learning rate of 0.00001, a single epoch, and a batch size of 8. These hyperparameters yield an accuracy of 97.0276%.

Finally, the predictive ability of each model was evaluated using the ROCs and calculating the AUC. The results, are shown in Figure 6.

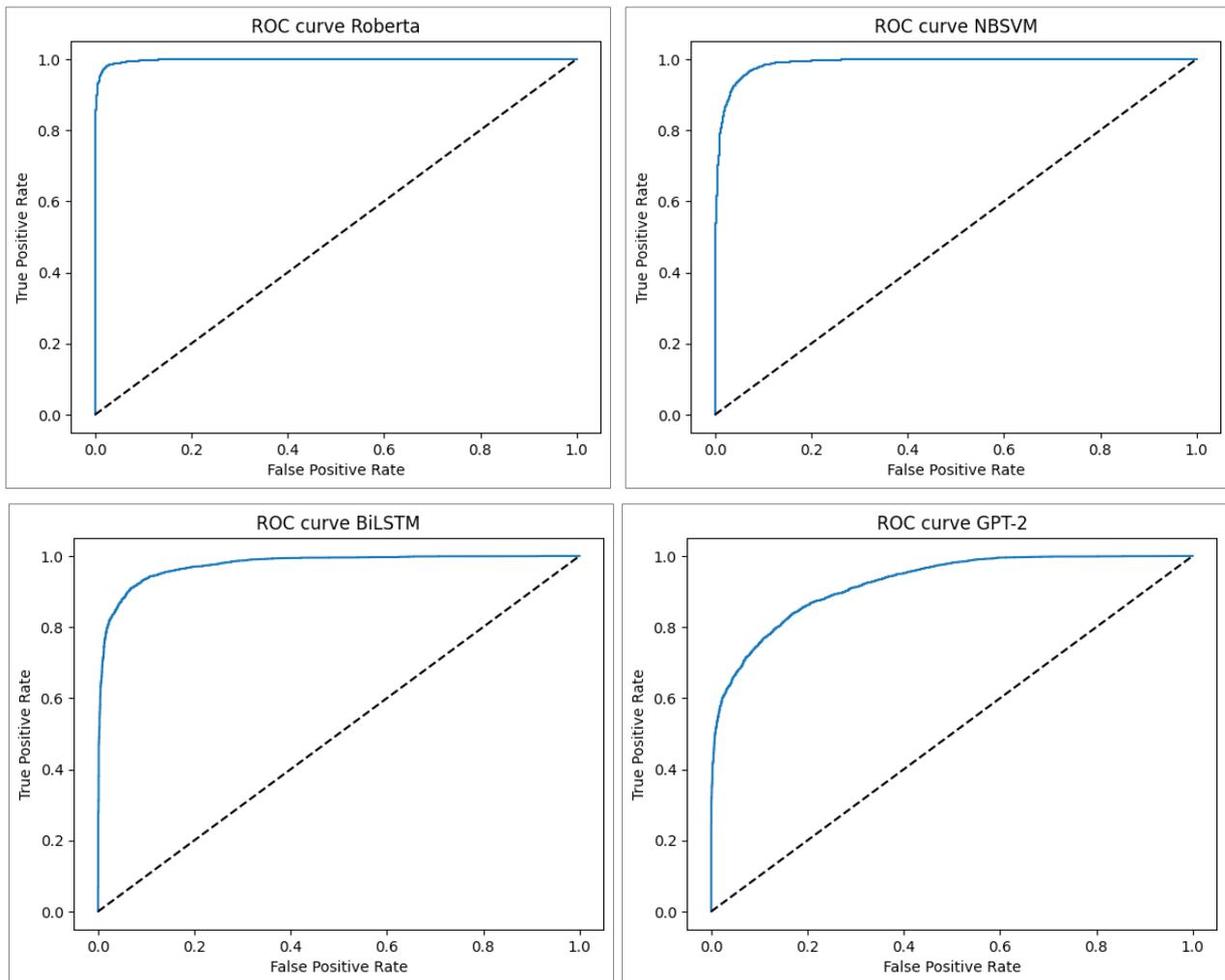


Figure 6. ROC cure in each distinct model

Figure 6 shows that the RoBERTa model demonstrated the best performance in detecting fake reviews, as evidenced by its AUC value of 0.9976, which among all, is the closest to 1. The NBSVM model achieved an area under the receiver operating characteristic (ROC) curve of 0.9888. Subsequently, the BiLSTM model demonstrated an AUC of 0.9753, while the GPT-2 model exhibited an AUC of 0.9226. The RoBERTa model is conclusively more accurate in detecting fake reviews than the others used as benchmarks in this study.

#### 4.2. Discussion

This study also includes a performance comparison with the previous study that compared related works that only used the Amazon dataset for evaluation. These performance details as shown in Table 7.

Table 7. Comparison of Model Performance

Model	Accuracy	Precision	Recall	F1-Score	AUC
FRDLSTM [26]	97.21%	96.00%	97.24%	96.61%	-
fakeRoBERTa [27]	96.64%	97.00%	97.00%	97.00%	96.62
BERT+SVM [31]	87.81%	-	-	88.00%	-
CNN-LSTM [33]	89.00%	90.00%	-	89.00%	-
GRU [34]	92.00%	-	-	-	-
<b>Propose Model (RoBERTa)</b>	<b>97.00%</b>	<b>97.00%</b>	<b>97.00%</b>	<b>97.00%</b>	<b>99.76%</b>

This study examined the GPT2, NBSVM, and RoBERTa, algorithms, which are similar to the ones investigated in the Joni Salminen et al. study on closeness [27]. It is worth noting that both studies utilized the same dataset, namely the Amazon Review Data (2018) [37]. Comparing the three models in this research paper and in Salminen et al. [27], confirms that the three models in this research paper and those in Salminen et al. [27] perform slightly worse than the

proposed model when measured by precision, recall, and F1-score, as shown in Table 7. In addition, when compared with the AUC metric, it is noteworthy that the proposed RoBERTa model, achieved the highest AUC score of 0.9976, while the NBSVM model closely followed with a score of 0.9888. The GPT-2 model's AUC score was determined to be 0.9226. In contrast, the three models examined in Salminen et al. [27] had notably lower scores in terms of AUC, with values of 0.696, 0.575, and 0.595, respectively.

A comparative study conducted by Shetgaonkar et al. [34] aimed to identify CGs using sentiment analysis and deep learning neural networks such as GRU, Bi-LSTM, and LSTM. The study used a dataset of mobile phone products from GitHub, which already had positive and negative reviews categorized into separate files. To prepare the data, that study proposed a model that tokenized the text reviews into smaller words or lines. Additionally, the study utilized the GloVe word embedding approach to provide clear word contexts throughout the entire text corpus. After a performance test on the proposed model, it was discovered that the RoBERT and BiLSTM models in this study had higher accuracy when compared to the Shetgaonkar et al. [34] study. Specifically, the RoBERT and BiLSTM models achieved accuracy values of 97% and 92%, respectively, which is higher than the 91% accuracy achieved by the Bi-LSTM model in the Shetgaonkar et al. [34] study.

However, the model proposed in the present study has a lower performance when compared with the FRDLSTM model of Qayyum et al. [26]. The FRDLSTM model applied the DCWR algorithm to compute deep features, used PCA to reduce the feature space, and identified fake reviews by training the Bi-LSTM model, which presents an interesting challenge for future work.

## 5. Conclusion

Checking if the reviews and opinions from customers are fake. It is considered a challenge. Because it allows customers to make a decision to buy products and services that are not mistaken. It also helps good business owners prevent fake customer reviews and opinions that damage the reputation of their business. In this study, the effectiveness of different sentiment analysis techniques is compared. These methods include RoBERTa, BiLSTM, GPT-2, NBSVM, etc. Amazon Review Data (2018) is the used dataset. The RoBERTa model, according to the results, performs the best overall. the GPT-2 model, the NBSVM model, the BiLSTM model, and the model with an overall efficiency of 97%, 82%, 95%, and 92%, respectively. However, the fake customer reviews and feedback data used in this study are computer-generated. In the future, we should continue to test human-generated fake customer reviews and opinions.

## 6. Declarations

### 6.1. Author Contributions

Conceptualization, L.B. and M.P.; methodology, L.B. and M.P.; software, M.P.; validation, L.B., M.P., and K.T.; formal analysis, M.P.; investigation, L.B., M.P., and K.T.; resources, M.P.; data curation, M.P.; writing—original draft preparation, K.T.; writing—review and editing, L.B. and K.T.; visualization, L.B. and K.T.; supervision, L.B. and K.T.; project administration, M.P.; funding acquisition, K.T. and L.B. All authors have read and agreed to the published version of the manuscript.

### 6.2. Data Availability Statement

Data sharing is not applicable to this article.

### 6.3. Funding

The authors received financial support for publication from the Walailak University, Thailand.

### 6.4. Acknowledgements

We would like to express our sincere gratitude to King Mongkut's Institute of Technology Ladkrabang and Walailak University for their invaluable support in aiding our research efforts. The study's findings are expected to have a significant influence on the current corpus of research in the field.

### 6.5. Institutional Review Board Statement

Not applicable.

### 6.6. Informed Consent Statement

Not applicable.

### 6.7. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## 7. References

- [1] Santos, K. E. S. (2020). Online Marketing: Benefits and Difficulties to online Business Sellers. *International Journal of Advanced Engineering Research and Science*, 7(3), 159–163. doi:10.22161/ijaers.73.27.
- [2] Zhu, L., Li, H., Wang, F. K., He, W., & Tian, Z. (2020). How online reviews affect purchase intention: a new model based on the stimulus-organism-response (S-O-R) framework. *Aslib Journal of Information Management*, 72(4), 463–488. doi:10.1108/AJIM-11-2019-0308.
- [3] Mehmeti-Bajrami, S., Qerimi, F., & Qerimi, A. (2022). The Impact of Digital Marketing vs. Traditional Marketing on Consumer Buying Behavior. *HighTech and Innovation Journal*, 3(3), 326–340. doi:10.28991/HIJ-2022-03-03-08.
- [4] Jain, P. K., Pamula, R., & Srivastava, G. (2021). A systematic literature review on machine learning applications for consumer sentiment analysis using online reviews. *Computer Science Review*, 41, 100413. doi:10.1016/j.cosrev.2021.100413.
- [5] Hu, S., Kumar, A., Al-Turjman, F., Gupta, S., Seth, S., & Shubham. (2020). Reviewer Credibility and Sentiment Analysis Based User Profile Modelling for Online Product Recommendation. *IEEE Access*, 8, 26172–26189. doi:10.1109/ACCESS.2020.2971087.
- [6] Van Vu, D., Tran, G. N., & Van Nguyen, C. (2022). Digital Transformation, Student Satisfaction, Word of Mouth and Online Learning Intention in Vietnam. *Emerging Science Journal*, 6, 40–54. doi:10.28991/ESJ-2022-SIED-04.
- [7] 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. doi:10.1109/ACCESS.2021.3075573.
- [8] Alghamdi, J., Lin, Y., & Luo, S. (2022). A Comparative Study of Machine Learning and Deep Learning Techniques for Fake News Detection. *Information (Switzerland)*, 13(12), 576. doi:10.3390/info13120576.
- [9] Haque, I., Alim, M., Alam, M., Nawshin, S., Noori, S. R. H., & Habib, M. T. (2022). Analysis of recognition performance of plant leaf diseases based on machine vision techniques. *Journal of Human, Earth, and Future*, 3(1), 129–137. doi:10.28991/HEF-2022-03-01-09.
- [10] Nandwani, P., & Verma, R. (2021). A review on sentiment analysis and emotion detection from text. *Social Network Analysis and Mining*, 11(1), 81. doi:10.1007/s13278-021-00776-6.
- [11] Dadhich, A., & Thankachan, B. (2021). Sentiment Analysis of Amazon Product Reviews Using Hybrid Rule-based Approach. *International Journal of Engineering and Manufacturing*, 11(2), 40–52. doi:10.5815/ijem.2021.02.04.
- [12] Lajkó, M., Csuvik, V., & Vidács, L. (2022). Towards JavaScript program repair with generative pre-trained transformer (GPT-2). *Proceedings of the Third International Workshop on Automated Program Repair*. doi:10.1145/3524459.3527350.
- [13] Golpour, P., Ghayour-Mobarhan, M., Saki, A., Esmaily, H., Taghipour, A., Tajfard, M., Ghazizadeh, H., Moohebbati, M., & Ferns, G. A. (2020). Comparison of support vector machine, naïve bayes and logistic regression for assessing the necessity for coronary angiography. *International Journal of Environmental Research and Public Health*, 17(18), 1–9. doi:10.3390/ijerph17186449.
- [14] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized Bert pretraining approach. *arXiv preprint arXiv:1907.11692*. doi:10.48550/arXiv.1907.11692.
- [15] Ghosh, L., Saha, S., & Konar, A. (2020). Bi-directional Long Short-Term Memory model to analyze psychological effects on gamers. *Applied Soft Computing Journal*, 95, 106573. doi:10.1016/j.asoc.2020.106573.
- [16] Mahyoob, M., Algaraady, J., & Alrahaili, M. (2020). Linguistic-Based Detection of Fake News in Social Media. *International Journal of English Linguistics*, 11(1), 99. doi:10.5539/ijel.v11n1p99.
- [17] Koli, R., & Redekar, S. (2023). A Review on Sentiment Analysis Methodologies, Practices and Applications with Machine Learning. *International Journal of Computer Science and Mobile Computing*, 12(6), 64–70. doi:10.47760/ijcsmc.2023.v12i06.007.
- [18] Kotelnikova, A. V. (2020). Comparison of Deep Learning and Rule-based Method for the Sentiment Analysis Task. *2020 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon)*, Vladivostok, Russia. doi:10.1109/fareastcon50210.2020.9271333.
- [19] Asghar, M. Z., Khan, A., Ahmad, S., Qasim, M., & Khan, I. A. (2017). Lexicon-enhanced sentiment analysis framework using rule-based classification scheme. *PLoS ONE*, 12(2), 171649. doi:10.1371/journal.pone.0171649.
- [20] Qader, W. A., Ameen, M. M., & Ahmed, B. I. (2019). An Overview of Bag of Words; Importance, Implementation, Applications, and Challenges. *2019 International Engineering Conference (IEC)*. doi:10.1109/iec47844.2019.8950616.
- [21] Cahyani, D. E., & Patasik, I. (2021). Performance comparison of TF-IDF and word2vec models for emotion text classification. *Bulletin of Electrical Engineering and Informatics*, 10(5), 2780–2788. doi:10.11591/eei.v10i5.3157.
- [22] Hamayel, M. J., & Owda, A. Y. (2021). A Novel Cryptocurrency Price Prediction Model Using GRU, LSTM and bi-LSTM Machine Learning Algorithms. *AI (Switzerland)*, 2(4), 477–496. doi:10.3390/ai2040030.

- [23] Alammar, J. (2018). The Illustrated GPT-2 (Visualizing Transformer Language Models). Available online: <http://jalammar.github.io/illustrated-gpt2/> (accessed on April 2023).
- [24] Acheampong, F. A., Nunoo-Mensah, H., & Chen, W. (2021). Transformer models for text-based emotion detection: a review of BERT-based approaches. *Artificial Intelligence Review*, 54(8), 5789–5829. doi:10.1007/s10462-021-09958-2.
- [25] Vidanagama, D. U., Silva, A. T. P., & Karunananda, A. S. (2022). Ontology based sentiment analysis for fake review detection. *Expert Systems with Applications*, 206, 117869. doi:10.1016/j.eswa.2022.117869.
- [26] Qayyum, H., Ali, F., Nawaz, M., & Nazir, T. (2023). FRD-LSTM: a novel technique for fake reviews detection using DCWR with the Bi-LSTM method. *Multimedia Tools and Applications*, 82(20), 31505–31519. doi:10.1007/s11042-023-15098-2.
- [27] Salminen, J., Kandpal, C., Kamel, A. M., Jung, S., & Jansen, B. J. (2022). Creating and detecting fake reviews of online products. *Journal of Retailing and Consumer Services*, 64, 102771. doi:10.1016/j.jretconser.2021.102771.
- [28] Yao, J., Zheng, Y., & Jiang, H. (2021). An Ensemble Model for Fake Online Review Detection Based on Data Resampling, Feature Pruning, and Parameter Optimization. *IEEE Access*, 9, 16914–16927. doi:10.1109/ACCESS.2021.3051174.
- [29] Rathore, P., Soni, J., Prabakar, N., Palaniswami, M., & Santi, P. (2021). Identifying Groups of Fake Reviewers Using a Semisupervised Approach. *IEEE Transactions on Computational Social Systems*, 8(6), 1369–1378. doi:10.1109/TCSS.2021.3085406.
- [30] Tufail, H., Ashraf, M. U., Alsubhi, K., & Aljahdali, H. M. (2022). The Effect of Fake Reviews on e-Commerce during and after Covid-19 Pandemic: SKL-Based Fake Reviews Detection. *IEEE Access*, 10, 25555–25564. doi:10.1109/ACCESS.2022.3152806.
- [31] Mir, A. Q., Khan, F. Y., & Chishti, M. A. (2023). Online Fake Review Detection Using Supervised Machine Learning And Bert Model. *arXiv preprint arXiv:2301.03225*. doi:10.48550/arXiv.2301.03225.
- [32] S, M. (2023). Detection of Fake Product Review using Machine Learning Techniques. *International Journal for Research in Applied Science and Engineering Technology*, 11(5), 1332–1338. doi:10.22214/ijraset.2023.51761.
- [33] Alsubari, S. N., Deshmukh, S. N., Al-Adhaileh, M. H., Alsaade, F. W., & Aldhyani, T. H. H. (2021). Development of Integrated Neural Network Model for Identification of Fake Reviews in E-Commerce Using Multidomain Datasets. *Applied Bionics and Biomechanics*, 2021, 1–11. doi:10.1155/2021/5522574.
- [34] Shetgaonkar, P., Rodrigues, J. T., Aswale, S., Gonsalves, V. L. K., Rodrigues, J. C., & Naik, A. (2021). Fake Review Detection Using Sentiment Analysis and Deep Learning. 2021 International Conference on Technological Advancements and Innovations (ICTAI), Tashkent, Uzbekistan. doi:10.1109/ictai53825.2021.9673375.
- [35] Elmogy, A. M., Tariq, U., Mohammed, A., & Ibrahim, A. (2021). Fake Reviews Detection using Supervised Machine Learning. *International Journal of Advanced Computer Science and Applications*, 12(1), 601–606. doi:10.14569/ijacsa.2021.0120169.
- [36] Nagi Alsubari, S., N. Deshmukh, S., Abdullah Alqarni, A., Alsharif, N., H. H. Aldhyani, T., Waselallah Alsaade, F., & I. Khalaf, O. (2022). Data Analytics for the Identification of Fake Reviews Using Supervised Learning. *Computers, Materials & Continua*, 70(2), 3189–3204. doi:10.32604/cmc.2022.019625.
- [37] Ni, J., Li, & McAuley. (2018). Empirical Methods in Natural Language Processing (EMNLP). Amazon Review Data. Available online: [https://cseweb.ucsd.edu/~jmcauley/datasets/amazon\\_v2](https://cseweb.ucsd.edu/~jmcauley/datasets/amazon_v2) (accessed on March 2023).