Fake Review Detection of E-commerce Using GNN

Submitted in partial fulfillment of the requirements of the degree of

BACHELOR OF COMPUTER ENGINEERING

by

Yash Pol (19102068)

Akshen Dhami (18102032)

Amish Nandu (18102048)

Sumati Hans (18102028)

Guide

Prof. Bharti Khemani

Department of Computer Engineering

A. P. SHAH INSTITUTE OF TECHNOLOGY, THANE

2022-2023

A. P. SHAH INSTITUTE OF TECHNOLOGY, THANE

CERTIFICATE

This is to certify that the project entitled "**Fake Review Detection of E-commerce Using GNN**" is a bonafide work of **"Yash Pol (19102068), Akshen Dhami (18102032), Amish Nandu (18102048), Sumati Hans (18102028)"** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **Bachelor of Engineering** in **Computer Engineering.**

Guide Prof. Bharti Khemani	Project Coordinator Prof. Rushikesh Nikam
Head of Department Prof. Sachin Malave	Principal Dr. Uttam Kolekar

Date:

A. P. SHAH INSTITUTE OF TECHNOLOGY, THANE

Project Report Approval for B.E.

This project report for Sem-VIII entitled **Fake Review Detection of E-commerce Using GNN** by ***Yash Pol (19102068), Akshen Dhami (18102032), Amish Nandu (18102048), Sumati Hans (18102028)*** is approved for the degree of ***Bachelor of Engineering*** in ***Computer Engineering***, ***2022-23***.

Examiner Name Signature

1. ____________________________

2. ____________________________

Date:

Place:

Declaration

We declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the sources. I also declare that I have adhered to all academic honesty and integrity principles and have not misrepresented, fabricated, or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and canalso evoke penal action from the sources which have thus not been properly citedor from whom proper permission has not been taken when needed.

(Yash Pol 19102068)

(Akshen Dhami18102032)

(Amish Nandu 18102048)

(Sumati Hans 18102028)

Date:

Abstract

Dishonest opinions, usually called fake reviews, are utilized to mislead individuals and have recently gained relevance. This is a result of the swift growth in Internet marketing transactions, including buying and selling. Customers can write reviews and comments on their purchased products or services using e-commerce. Before making a purchasing decision, new buyers frequently read the website's reviews or comments. How new people, however, can tell real evaluations from false ones, which later mislead customers, cause losses, and damage businesses' reputations, is now an issue. The current study aims to create an intelligent system that can identify fake reviews on e-commerce platforms utilizing the n-grams of the review text and the reviewer's sentiment scores. The proposed methodology used in this work employed a Count Vectorization approach for feature extraction and their representation, together with a standard fake e-commerce review dataset for experimentation and data pretreatment procedures. N-grams from review texts were entered into the built models for detection and classification to determine if they were fraudulent or real. However, a dataset obtained from the Amazon website was used to train and test three different supervised machine-learning techniques, naive Bayes (NB), support vector machine (SVM), logistic regression (LR,) and three deep-learning techniques, MLP Classifier, Graph Convolutional Networks (GCN) and Long Short-Term Memory (LSTM). We construct a graph representation of the review corpus, where each node corresponds to a word in the review and edges represent the co-occurrence of words. The GCN is then used to learn a representation of the graph, which captures the structural and semantic relationships among the words. The LSTM is then used to capture the review’s temporal dependencies and classify them as fake or genuine.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Abbreviation

AI	Artificial Learning
ML	Machine Learning
DL	Deep Learning
SVM	Support Vector Machine
LR	Logistic Regression
LSTM	Long-Short term Memory
MLP Classifiers	Multi-layer Perceptron classifier
GCN	Graph Convolution Network
GNN	Graph Neural Network
CNN	Convolution Neural Network
NLP	Natural Language Processing
GAT	Graph Attention Network
KMGCN	Knowledge-driven Multimodal Graph Convolution
HIGNN	Hierarchical Bipartite Graph Neural Network
FRD	Fake Review Detection
DAGA-NN	Domain-adversarial and graph-attention neural network
IRGNN	Item Relationship Graph Neural Networks
RNN	Recurrent neural network
NB	Naïve Bayes

CHAPTER 1

Introduction

Fake review is a serious issue in online stores and greatly impacts buyers' decisions. Due to their capacity to simulate the intricate relationships between various reviews and reviewers, Graph Neural Networks (GNNs) have become an effective method for identifying fake reviews.

Each review and reviewer are represented as nodes in a graph in GNN-based false review detection, and the edges between them capture the relationships between them, including the similarity of their content or the frequency of their interactions. Using several graph convolutional layers, GNNs then learn to transmit information between nodes in the graph to spot patterns that are suggestive of fraudulent review.

GNN-based fake review detection has demonstrated promising results in identifying isolated instances of fake reviews and coordinated campaigns in which a single group posts numerous fake reviews. It might greatly increase the validity of online reviews and boost consumer confidence in online marketplaces.

A more recent study proposed a new GNN-based method for fake review detection that takes into account both the content of reviews and the graph structure. They constructed a heterogeneous graph using the Yelp Challenge dataset, where nodes represent reviewers, reviews, and products, and edges represent the relationships between them. In addition to the standard graph convolutional layers, their model also includes attention mechanisms that allow it to focus on the most relevant parts of the graph for each review. The authors compared their method to several state-of-the-art approaches on the Yelp Challenge dataset and found that it achieved the best performance in terms of both accuracy and F1 score[2]. Other studies have also explored the use of GNNs for fake review detection in various contexts. For example, A paper proposed a method that uses a graph attention network (GAT), another type of GNN, to analyze the reviews and reviewers in a graph constructed from TripAdvisor, a travel review platform. Their results showed that their method achieved higher accuracy and F1 score than several baseline methods and also extended this work by incorporating user features, such as age and gender, into the graph and showed that their method improved the performance further[6].

In conclusion, GNNs have shown great potential in detecting fake reviews by leveraging the graph structure of review datasets. These methods have demonstrated superior performance over traditional machine learning models and neural networks that do not take into account the graph structure. As the problem of fake reviews continues to be a pressing issue for online platforms, GNN-based methods are likely to become increasingly important for ensuring the integrity of online reviews.

CHAPTER 2

Literature Survey

Fake review detection is a growing field of research that aims to develop automated techniques for identifying fraudulent or deceptive reviews on online platforms. The literature on this topic includes various approaches such as machine learning, natural language processing, and graph neural networks.

In the year 2020, HiGNN (Hierarchical bipartite Graph Neural Network) obtained the hierarchical structure effectively and efficiently and produce a notable improvement when compared to state-of-the-art baselines in Taobao (one of the biggest real-world e-commerce platforms)[1]. Data from two publicly released real-world social network databases: Pheme and Weibo, were collected in 2020 to verify the efficiency of the suggested system KMGCN (Knowledge-driven Multimodal Graph Convolution)[2]. In the year 2021, Research on fake review detection Methods: A state-of-the-art review involves summarizing the key issues of the current fake review text detection research from the standpoint of the technical route of spam review detection research. This includes introducing and comparing false feature design, recognition models, and methods, and then summarizing and discussing the current problems and issues in research[3]. “Development of Integrated Neural Network Model for Identification of Fake Review in E-Commerce Using Multidomain Datasets” published in the year 2021, identifies and categorizes review content as being fraudulent or authentic, offering a hyper neural network model composed of convolutional neural network and long short-term memory (CNN-LSTM) approaches[4]. In the year 2021, Fake review detection on Online E-commerce Platforms: a systematic literature review (Himangshu Paul, Alexander Nikolaev), research summarizes the contributions of the most recent FRD studies, this systematic literature review demonstrated how the FRD tactics continually become more complicated in response to the changing spamming techniques. It classified FRD modeling methods according to the researchers' hypotheses about how fake reviews are created and how to review spammers' acts[5]. Improving fake news detection with domain-adversarial and graph-attention neural network paper was published in the year 2021. Where they used DAGA-NN. The results of the experiment on two genuine corpora demonstrated that DAGA-NN outperformed cutting-edge baselines[6]. In the year 2022 Online Spam Review Detection: A Survey of Literature, displayed four freely accessible datasets (TripAdvisor, Amazon, Yelp, Dianping). Modern GCN-based algorithms for detecting spam reviews in the actual world are used: MNCN, FdGars, GCNN, and GAS.[7]. In the Year 2022, Creating and detecting fake reviews of online products: Proves that machine learning classifiers are far much better in this area than others, recognizing evaluations produced by other machines with virtually flawless accuracy. This suggests that in the fight against fake reviews, "machines can fight machines." The dataset used had a total of around 400 written fake reviews by crowd workers recruited via Amazon Mechanical Turk[8]. In the year 2022 Data analytics for the identification of fake reviews using a gold standard dataset developed by Ott et al involves applying supervised machine-learning algorithms, including naive Bayes, support vector machines, random forests, and adaptive boost, were investigated and put into practice to identify fake reviews. Utilizing the TF-IDF approach, features were extracted. When analyzing the classification outcomes of tests, it was shown that the random forest classifier performed better than other classifiers in identifying fake reviews, obtaining a 95% accuracy rate and F1-score metric. Higher sensitivity metrics (94%) were reached by the Adaboost classifier[9]. In the year 2022, Item Relationship Graph Neural Networks for E-Commerce, IRGNN, a GNN-based framework was implemented. IRGNN automatically learns topological features and relational dependencies in multi-hop neighborhoods to improve the quality of item relationship prediction rather than merely depending on the item content information[10]. In the year 2022, DC-GNN: Decoupled Graph Neural Networks for Improving and Accelerating Large-Scale E-commerce Retrieval. To address the trade-off between model performance and training efficiency, DC-GNN was proposed, which decouples the conventional GNN-based CTR prediction paradigm for large-scale retrieval. Deep aggregation and multi-tasks-based pre-train stages both significantly enhance model performance. In essence, DC-GNN offered a workable substitute or complementary framework for extensive E-commerce retrieval[11]. In the year 2022, A Survey on the Use of Graph Convolutional Networks for Combating Fake News demonstrates the advantages of graph-based representations and GCNs over simple CNNs or RNNs in graph and node classification tasks. The graphs better capture the latent relations between concepts in the news text, between the users that spread the news, or between users and the texts they share[12].

Table 1: Survey of Literature

Title (Year)	Description	Research Gap
[1] (2020)	Compared to the most recent baselines in Taobao, HiGNN (Hierarchical bipartite Graph Neural Network) achieved the hierarchical structure effectively and efficiently.	While Graph Neural Network (GNN) based state-of-the-art techniques have made significant progress in leveraging high-order connections, non-linear interactions, and hierarchical representations on user-item bipartite graphs to optimize collaborative filtering, they fall short in demonstrating their potential in large-scale e-commerce settings.
[2] (2020)	In 2020, information from the two publicly accessible real-world social network databases Pheme and Weibo was gathered to assess the effectiveness of the proposed system KMGCN (Knowledge-driven Multimodal Graph Convolution).	In this paper, the CNN algorithm cannot work more efficiently than SVTMs or GRU on two datasets, because it ignores the long-range semantic relations among words and local feature is not enough to make a judgment for a post.
[3] (2021)	The introduction and comparison of fake feature design, recognition models, and methodologies, as well as a summary and discussion of the present research challenges, are all included in this.	Covers research that uses fake review text as identification target, Review Metadata Techniques for feature analysis such as It is impossible to find fake reviewers whose behavior is not much different from normal reviewers, and Features based on manual observation are subjective.

Title (Year)	Description	Research Gap
[4] (2021)	Offers a hyper neural network model consisting of convolutional neural network and long short-term memory (CNN-LSTM) techniques that can identify and classify review content as false or authentic.	According to the literature review of fake review detection methods, no research work has used the same datasets in a cross-domain experiment. Thus, they are unable to make comparative analyses with cross-domain datasets.
[5] (2021)	Summarized the contributions of the most current FRD studies and showed how the FRD strategies are always evolving to keep up with the evolving spamming methods. It categorized FRD modeling techniques by the researchers' theories regarding the production of fraudulent reviews and the behavior of review spammers.	Recently developed algorithms that use textual, behavioral, temporal, rating-based, and graph-based features in aggregation tend to be slow and not scalable; most reported approaches are designed to cater to a particular platform, which makes their performance domain-specific and less flexible; and all these approaches fail to distinguish between the malicious review and truthful atypical review, which leads to unnecessary harassment of truthful reviewers.
[6] (2021)	To learn the domain-invariant features and identify fake news across events and domains, DAGANN uses the notion of the domain adversarial network to introduce a domain discriminator and play a minimax game with the feature extractor. The experiment's findings on two real corpora showed that DAGA-NN outperformed state-of-the-art baselines.	Firstly, different fake newsmakers may handle the original content in different ways. As a result, a news story’s true and fake content may be mixed to different degrees. This gives rise to the problem of multiple classifications in fake news identification.

Title (Year)	Description	Research Gap
[7] (2022)	Four freely available datasets were displayed in A Survey of Literature (TripAdvisor, Amazon, Yelp, Dianping). Where modern GCN-based algorithms, such as MNCN, FdGars, GCNN, and GAS, are used to find spam reviews in the real world.	In this paper, there is a lack of state-of-art GCN-based techniques for real-world spam review detection.
[8] (2022)	Demonstrates that machine learning classifiers are significantly more accurate than other classifiers in this domain, detecting ratings made by other machines with almost perfect precision. This implies that "machines can fight machines" in the battle against fake reviews.	One limitation is that the original Amazon review dataset might already involve an unknown number of fake reviews. If so, this would result in a bias for the language model we applied. Unfortunately, we cannot know whether a given review in the dataset is undeniably truthful.
[9] (2022)	Utilize supervised machine-learning techniques to identify fake reviews. These techniques include naive Bayes, support vector machines, random forests, and adaptive boosts. The random forest classifier outperformed other classifiers in recognizing fraudulent reviews, achieving a 95% accuracy rate.	The use of pre-trained models assists feature and concept extraction, but the association of features with concepts still requires training for fine-tuning the different hyper-parameters.

Title (Year)	Description	Research Gap
[10] (2022)	A GNN-based framework called IRGNN was used. Instead of relying solely on the knowledge of the item's content, IRGNN automatically learns topological features and relational dependencies in multi-hop neighborhoods to enhance the quality of item relationship prediction.	The current model may not be able to handle dynamic changes in item relationships that occur over time due to daily transactions, as it relies on the assumption of static training data.
[11] (2022)	The DC-GNN, which decouples the traditional GNNs-based CTR prediction paradigm for large-scale retrieval into three stages—pre-train, deep aggregation, and two-tower CTR prediction—was developed to address the trade-off between model performance and training efficiency.	Modifying graph operators in a simplistic manner can lead to a decline in the graph representation capability, which is particularly evident in attribute graphs.
[12] (2022)	The different GCN techniques presented in this article demonstrate the advantages of graph-based representations and GCNs over simple CNNs or RNNs in graph and node classification tasks. The graphs better capture the latent relations between concepts in the news text, between the users that spread the news, or between users and the texts they share.	GCN-based methods suffer from high complexity even when the eigenvectors are pre-computed or rank approximations of the eigenvalue decomposition are employed, the complexity is high, especially for large-scale graphs. Another major issue that is evident in all works is the large number of hyper-parameters that must be learned in each case.

CHAPTER 3

Limitation of Existing system

- **Lack of labeled data**: GNNs require a large amount of labeled data to achieve good performance. However, labeled data are scarce for fake review detection, which makes it difficult to train GNN models.

- **Limited generalization**: GNNs are effective in detecting fake reviews within a specific domain, such as restaurant reviews or hotel reviews. However, they may not generalize well to other domains, as the underlying graph structure may be different.

- **Scalability Issues**: GNNs require significant computational resources, which can make it challenging to scale the models to larger datasets. Additionally, the graph structure may become too complex to be efficiently processed by the GNN model.

- **Adversarial attacks**: Adversarial attacks can be used to manipulate the graph structure or the review text to evade detection by GNN-based models. These attacks can be difficult to detect and mitigate.

- **Limited interpretability**: GNNs can be difficult to interpret, as the model's decisions are based on the complex interactions between nodes in the graph. This makes it challenging to understand why a particular review was flagged as fake.

Overall, while GNNs show promise for fake review detection, there are still several challenges that need to be addressed to achieve robust and reliable detection.

CHAPTER 4

Problem Statement, Objectives, and Scope

4.1 Problem Statement

To implement the GNN algorithm for e-commerce fake review identification, where fake review detection refers to the act of recognizing and misleading online reviews. In order to prevent the review ecosystem from being used for fraudulent or promotional purposes, companies typically, take this step. To address these problems and protect the credibility of the online review system, fake review identification is required.

4.2 Objectives

The project's objective is to develop a model using Amazon’s Fake Review Detection feature.

1. To develop GNN architecture by defining the node and edge attributes, creating a graph representation of the data, and designing a neural network architecture for it.
2. To train the model by using a good optimization algorithm, we will modify the neural network's weights during training so that it can figure out how to anticipate the intended result for each input graph.
3. To assess the model's performance after the training, by testing it on a different set of data. This entails calculating metrics like accuracy. To comprehend the model's prediction process better, we will apply visualization tools.

4.3 Scope

The scope of fake review detection is quite broad, as it can be applied to many different domains, below we list the scope for the E-Commerce domain.

- Fake Review detection in the e-commerce domain assists in the identification of fraudulent review which misguides the customers while purchasing the goods & uses them to boost the sales of a particular product or to discredit a competitor's product.
- Detecting fake reviews spread through various social media platforms like Amazon.
- As more companies go online and as online reviews become more significant in decision-making, the scope of fraudulent review identification is continuously expanding. To safeguard customers and uphold the integrity of online review systems, there is a rising need for efficient and precise fraudulent review identification technologies.

CHAPTER 5

Proposed System

5.1 Proposed system overview

As the number of online reviews rises and more customers rely on reviews to guide their purchases, fake review detection systems are becoming more and more crucial. In this project, we aim to use Machine Learning and mainly deep learning method for fake review detection and see which algorithms model is the best.

5.2 Architecture Diagram

An architectural diagram is a visual representation that maps out the physical implementation of components of a software system. It shows the general structure of the software system and the associations, limitations, and boundaries between each element.

From the below Figure 5.2.1; Firstly, we gathered data containing the reviews from Kaggle & then we follow the below-given steps to identify wheatear the particular review is legitimate or fraudulent.

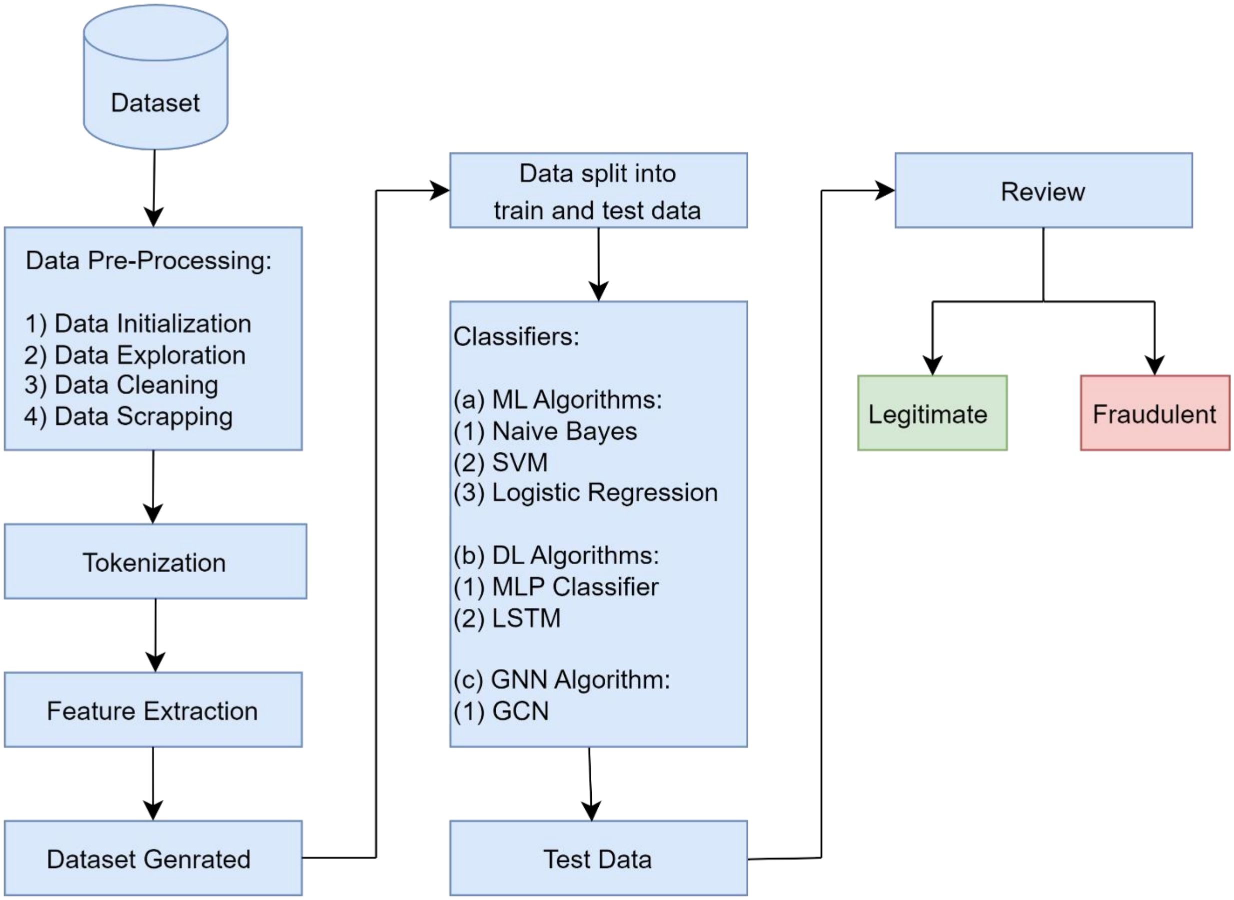


Figure 5.2.1: Architecture Diagram

The stages in the procedure are as follows:

1. **Data preprocessing:** Useful in converting raw data into an efficient and useful format. Here we are using various preprocessing methods like Data Cleaning, Data Exploration, Data Initialization, and Data Scraping.
   - **(a) <u>Data Initialization</u>:** We are initializing the data i.e., taking the raw data from the Kaggle dataset for further processing and algorithms for classification.
   - **(b) <u>Data Exploration</u>:** Here we are exploring the dataset to check the number of unwanted columns, and no. of null values and stop inwards as a part of feature extraction from the dataset.
   - **(c) <u>Data Cleaning</u>:** Data cleaning to clean the data i.e., to remove null values and unwanted rows or columns from the dataset.
   - **(d) <u>Data Scraping</u>:** To make new and efficient data that contains useful information we need several legitimate and fraudulent reviews.

2. **Tokenization:** After preprocessing, we move to tokenization, used to break up longer texts into words or short lines. We will use the new dataset to train and test the data and which is then followed by classification.
3. **Feature Extraction:** A process of selecting and extracting a subset of relevant features or variables from a larger set of raw data. It's done to reduce the dimensionality of the data and to simplify the problem, making it easier for algorithms to learn from the data and make accurate predictions. It's been done using various natural language processing (NLP) techniques, such as sentiment analysis, part-of-speech tagging, and named entity recognition. These features can help identify patterns and characteristics of legitimate and fraudulent reviews.
4. **Train and test the model**: Once the above procedures are finished, a new label dataset will be created, which we will use to train machine learning and deep learning models. The dataset should contain examples of both legitimate and fraudulent reviews, which will be used to train the machine learning & deep learning models, to recognize patterns in the data.
5. **Evaluate Model:** After the machine & deep learning model has been trained, it must be tested on a dataset to see how well it performs and how accurately it is. Metrics like precision, recall, and F1-score are been used for this. We will be using the following algorithms for our project which give precise accuracy for the dataset, which are as follows:
   (i) Naïve Bayes Classifier
   (ii) Support Vector Machine
   (iii) Logistic Regression
   (iv) MLP Classifier
   (v) LSTM
   (vi) GCN

6. **Deploy Model:** Once the machine & deep learning model performs well, it can be deployed to automatically detect fake reviews from the online platform.
7. **Monitor Results:** Finally, it is important to monitor the results of the fake review detection system over time to ensure its continued accuracy and effectiveness. This can include reviewing flagged reviews manually and adjusting models as needed.

In general, the procedure for identifying fraudulent reviews includes gathering review data, preprocessing the data, extracting out features, training and evaluating a machine learning model, deploying the model, and monitoring outcomes.

5.3 Methodology

In this part, the classification methodology is described. The process utilizes machine & deep learning to classify the dataset. Here in the classification process, the dataset collection phase is the initial step. Next, the training and testing of the dataset are executed after preprocessing, implementing features, and finally, the proposed system is defined by executing the classifiers. The strategy is centered on running a lot of experiments with the dataset and algorithms.

Here is a detailed explanation of the algorithm:

a) Machine Learning:

The development of algorithms and statistical models that enable computers to learn from data without being explicitly programmed is known as machine learning, and it is a subset of artificial intelligence. The foundation of machine learning is the premise that computers may learn from experience and gradually become better at a given activity. There are three main types of machine learning: Supervised Learning, Unsupervised Learning & Reinforcement Learning. Importantly, large datasets can be used to train machine learning algorithms, which helps them become more effective and accurate over time. This makes them valuable for a variety of real-world situations.

The following listed machine learning algorithms are applied:

(1) Naïve Bayes:

A popular probabilistic machine learning approach for classification tasks is naive Bayes. The Bayes theorem, which asserts that the likelihood of a hypothesis (in this case, a class label), is proportional to the likelihood that the data would support that hypothesis, is the foundation of the method[9]. Assuming that all features are independent of one another, which may not always be true in real-world data, gives Naive Bayes its "naive" name. Naive Bayes has been demonstrated to perform effectively in practice despite this presumption, especially for text classification problems.

(2) Support Vector Machine:

A common machine-learning approach for classification, regression, and outlier detection is called a Support Vector Machine (SVM). SVMs operate by locating the best hyperplane, with the highest possible margin, separating two classes in the feature space, where the margin is the separation between the hyperplane and the nearest data points for each class. By utilizing a kernel function to translate the input data into a higher-dimensional space where a linear hyperplane may separate the classes, SVMs can be utilized for both linear and nonlinear classification tasks. Radial basis function (RBF) kernels, polynomial kernels, and linear kernels are common types of kernel functions[9].

(3) Logistics Regression:

popular machine learning approach for binary classification tasks, where the objective is to estimate the likelihood of a binary result, is logistic regression (i.e., 0 or 1). The program simulates the link between one or more continuous or categorical independent variables and a binary dependent variable.

b) Deep Learning:

Artificial intelligence (AI) that uses deep learning is intended to learn from experience and get better over time without explicit programming. Deep neural networks are used in this area of machine learning to extract patterns from massive volumes of data. Deep learning methods use many layers of neural networks, which are referred to as "deep" networks. DL is particularly good at handling issues involving vast amounts of unstructured data, including text, audio, video, and image. The availability of big datasets, improvements in algorithmic techniques, and improvements in computer power have all contributed to the growing popularity of deep learning in recent years. TensorFlow, Keras, PyTorch, and Caffe are some of the deep learning frameworks that are most frequently used.

The following listed deep learning algorithms are applied:

(1) Multi-layer perceptron Classifier (MLP Classifier):

A form of artificial neural network known as a Multi-Layer Perceptron (MLP) Classifier is made up of many layers of perceptron, which are straightforward computing units that take a set of input values and return an output. The MLP Classifier is a sort of feedforward neural network, meaning that data only moves from the input layer to the output layer in one way. If the objective of a classification task is to predict a discrete output, such as a label or a category, the MLP Classifier can be employed. To learn to translate the input features to the appropriate output label, the MLP Classifier modifies the weights of the connections between the perceptron in each layer during training.

(2) **Long Short-Term Memory (LSTM):**

For handling sequential data, such as time series, text, or speech, recurrent neural networks (RNNs) of the Long Short-Term Memory (LSTM) type are used. In several tasks, including language modeling, machine translation, and speech recognition, long-term dependencies between the elements of a sequence must be captured by LSTMs. Memory cells, which can retain data for a long time, and gates, which can regulate the information flow into and out of the memory cells, make up LSTMs. The weights of the connections between the memory cells and the gates are changed during training to teach the LSTM how to process the input sequence and produce the desired output.

c) **Graph Neural Network (GNN):**

GNN stands for Graph Neural Network, a type of neural network architecture designed to operate on graph-structured data. Graphs are a type of data structure used to represent complex relationships between entities, where nodes represent entities and edges represent the relationships between them. GNNs work by sharing information across graph nodes, which enables them to learn representations of the nodes and their connections. This is accomplished by employing message-passing operations to combine data from nearby nodes in the graph. Many tasks, such as node classification, link prediction, and graph classification, have proved the efficacy of GNNs. They have been used in social network analysis, drug discovery, and recommendation systems, among other things.

Graph Convolutional Networks (GCNs), Graph Attention Networks (GATs), and GraphSAGE are a few examples of well-known GNN designs. TensorFlow, PyTorch, and DGL are just a few examples of deep learning frameworks that can be used to create these systems (Deep Graph Library).

In terms of identifying fraudulent reviews, GNNs have demonstrated promising results. Because fraudulent review frequently has identical wording and structure to legitimate ones, it can be difficult to identify fraudulent review from legitimate ones. Yet, by taking use of the review graph's underlying structure, GNNs can be efficient at spotting fake reviews.

Nodes in a review graph stand for review, and edges between them reflect connections such as similarities between reviews or the same individual reviewing several goods. In addition to being able to recognize patterns that are suggestive of fraudulent reviews, GNNs can be used to learn representations of these reviews and their relationships.

A GNN can be trained, for instance, to recognize groups of reviews that are strongly connected but weakly connected to the rest of the network. This could mean that a collection of fake reviews was produced by a single entity, like a fake review farm. A GNN can also be trained to recognize review that utilizes odd wording, such as overuse of superlatives or repeated words, which may be a sign that a review is fake.

GNNs have been employed in several studies for the detection of a fake review, with encouraging results. Nonetheless, there are currently ongoing efforts to increase the precision and robustness of GNN-based fake review detection algorithms because this is still a very hot area of research.

Under the model of GNN, we have implemented Graph Convolution Network (GCN).

(1) Graph Convolution Network (GCN):

Graph Convolutional Network is referred to as GCN. It is a kind of neural network that works with data that is graph-structured, such as social networks, citation networks, or chemical structures. One of the many uses for GCNs is the identification of fake reviews [12]. The objective of this task is to tell legitimate reviews from fraudulent ones. Through graph-modeling the connections between reviews, reviewers, and products, GCNs can aid in this job. A review, a reviewer, or a product is represented by each node in the graph, and the connections between them are shown by the edges connecting the nodes.

Information is spread among the graph's nodes through the GCN algorithm. Each cycle uses a convolution technique to update each node's features based on those of its neighbors. Up until the algorithm converges, the modified features are then used to update the features of the subsequent iteration.

The GCN can be trained to recognize patterns of behavior that are indicative of fraudulent reviews, such as numerous fake reviews coming from the same reviewer or reviews that are remarkably similar to one another. The GCN can capture more complex relationships between reviews, reviewers, and products than conventional machine learning models that work with flat feature vectors by taking advantage of the graph structure of the data.

The methodology is to use the GCN algorithm to learn a representation of the graph that captures the underlying semantics and relationships among the words. The GCN takes the graph representation of the review corpus as input and iteratively updates the node representations based on the features of neighboring nodes.

Let *H(l)* be the hidden representation of the graph at layer l, where *H(0)* is the input feature matrix, and *H(l+1)* is the hidden representation after layer l. The GCN algorithm can be represented as follows:

$$\boldsymbol{H}^{(\boldsymbol{l}+\mathbf{1})} = \boldsymbol{\sigma}(\widetilde{\boldsymbol{D}}^{-\frac{1}{2}}\widetilde{\boldsymbol{A}}\widetilde{\boldsymbol{D}}^{-\frac{1}{2}}\boldsymbol{H}^{(\boldsymbol{l})}\boldsymbol{W}^{(\boldsymbol{l})})$$

where D is the degree matrix of the graph, W(l) is the weight matrix at layer l, and σ is the activation function. The GCN algorithm updates the node representations at each layer by aggregating the features of neighboring nodes and applying a linear transformation[12].

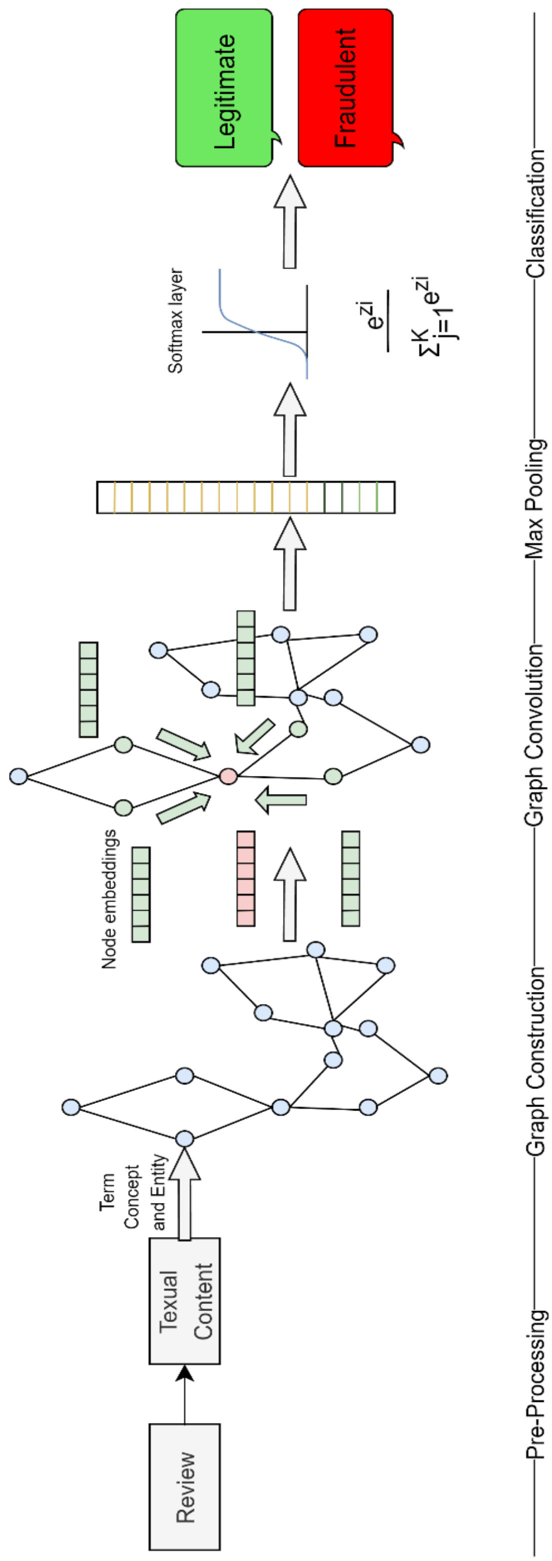


Figure 5.3.1 Working of GCN

CHAPTER 6

Experimental Setup

6.1 Software Requirements:

1. **Python:** Python is one of the widely used programming languages for building systems that indulge in Image Processing as well as Machine Learning. Python provides amazingly powerful libraries and tools that help us in achieving tasks efficiently.
   - **(a) NumPy:** This is a library that lets one perform simple image techniques such as flipping images, extracting features, and analyzing them.
   - **(b) Pandas:** Panda is an open-source library that is made mainly for working with relational or labeled data both easily and intuitively. It provides various data structures and operations for manipulating numerical data and time series. This library is built on top of the NumPy library.
2. **Jupyter:** Jupyter is a web-based interactive development environment for notebooks, code, and data. Its flexible interface allows users to configure and arrange workflows in data science, scientific computing, and machine learning. A modular design that invites extensions to expand and enrich functionality.
3. **PyCharm:** PyCharm is a dedicated Python Integrated Development Environment (IDE) providing a wide range of essential tools for Python developers, tightly integrated to create an environment for productive Python and web development.
4. Windows 11 Operating System.

6.2 Hardware Requirements:

- 8 cores processor
- 32 GB RAM
- 512 GB free space

CHAPTER 7

Project Plan

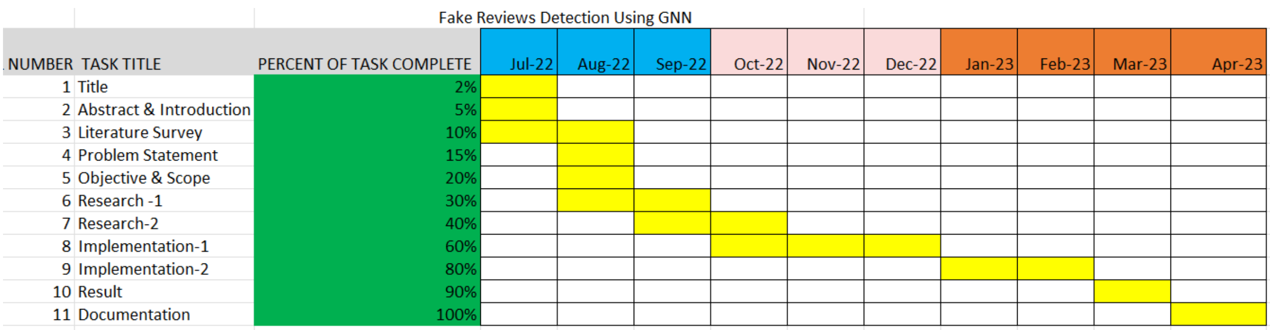


Figure 7.1.1 Gantt Chart

Referring to Figure 7.1.1, Our group began choosing a domain, and after narrowing it down to two, we followed our guide's advice and chose the AI-ML domain. Later, we took some time to choose two to three topics for the AI-ML domain, and after consulting with our guide once more, we decided on Fake review detection using GNN as our title. We then worked on our introduction and abstract. By the middle of July, the stated process was finished.

We reviewed various research publications from the second half of July through the end of August before defining our problem statement, objective, and scope. From the end of August to the beginning of October, we conducted research and learned about machine learning concepts, particularly deep learning, through research papers and videos on YouTube.

After establishing a solid foundation, we began working on the project in early October. By the end of March, we had successfully created a fake review detection model and deployed it on Streamlit, declaring that deep learning provides greater accuracy. In the end, we began to prepare our documentation, which we finished in April.

CHAPTER 8

Results

Our objective is to identify whether a specific review is legitimate or fraudulent. A comprehensive Amazon dataset was used considering various types of product reviews. We utilize natural language processing techniques, including sentiment analysis, topic modeling, and linguistic feature extraction, to extract relevant features from the text data. We have studied the various types of algorithms used for the detection of fraudulent reviews and moving on, our fake review detection system uses a combination of machine learning and deep learning algorithms to analyze the review on this platform and identify patterns of behavior that are consistent with the fake review. We compared by implementing multiple Machine Learning & Deep Learning models, such as Multinomial Naive Bayes, Support Vector Machine, Logistic Regression, MLP Classifier, LSTM & GCN as shown in Figure 8.1.1. The logistic Regression model for Machine Learning and Graph Convolutional Network (GCN) model a GNN-based Deep Learning has been generating the most consistent accuracy. Among Logistics Regression & GCN, GCN has the highest accuracy of 99.7% whereas Logistic Regression has around 85% accuracy.

8.1 Accuracy:

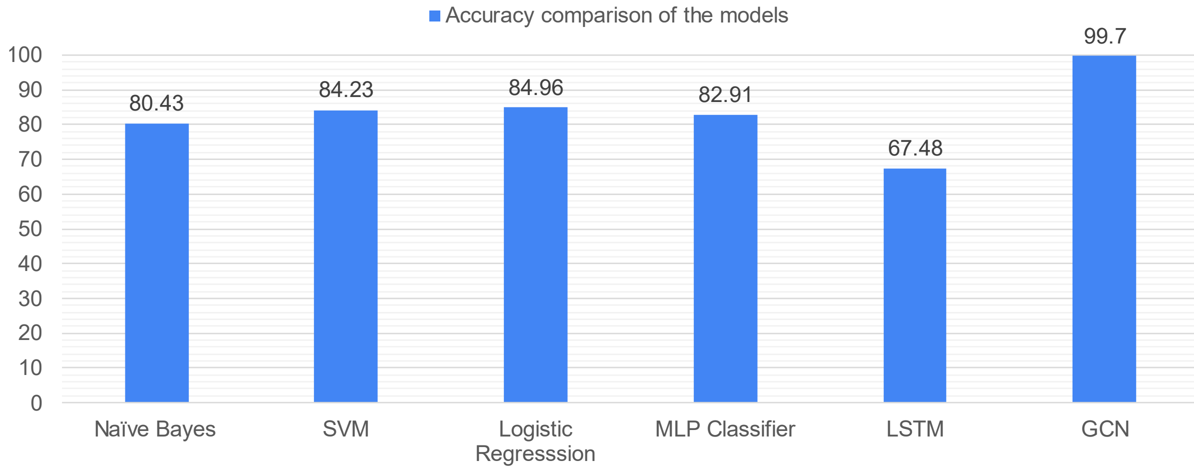


Figure 8.1.1: Accuracy comparison of the models

8.2 Implementation:

To deploy our model, we used Streamlit. It's an open-source Python library used to create web apps for data science and machine learning. It offers a simple interface for using Python code to build interactive dashboards and data visualizations. Without writing any HTML, CSS, or JavaScript, developers may construct web applications with Streamlit with just a few lines of Python code. A variety of pre-made UI elements, including sliders, buttons, and charts, are included in the library and may be used to quickly and easily create unique user interfaces.

Several of Streamlit's main characteristics are:

(1) **Real-time app updates**: Streamlit offers automated app updates, making it possible to make changes to the app's code without having to reload the page.

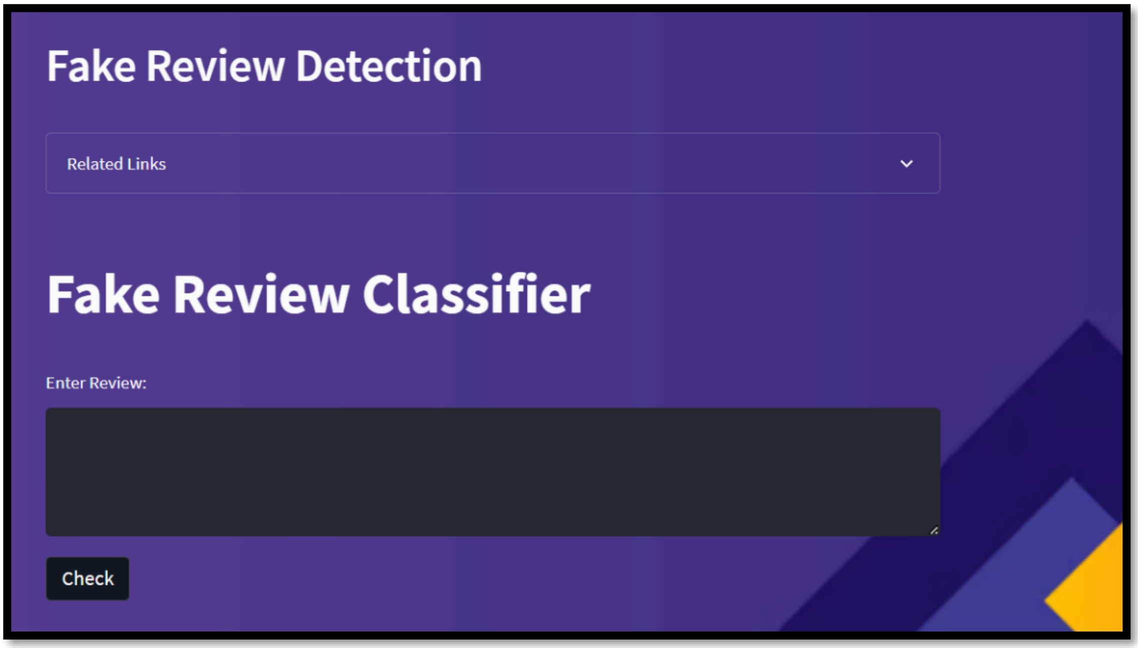
(2) **Interactive widgets:** Widgets that can be used to construct interactive user interfaces are included in Streamlit. These widgets include sliders, buttons, and drop-down menus.

(3) **Data visualization**: To view data in real time, Streamlit offers a variety of visualization tools, such as charts, graphs, and maps.

(4) **Sharing:** By deploying Streamlit apps to a range of hosting platforms like Heroku, Google Cloud, or Amazon, users can quickly share their creations with others.

Due to its simplicity of use and capacity to produce interactive web apps quickly, Streamlit has grown in favor among data scientists and machine learning experts.

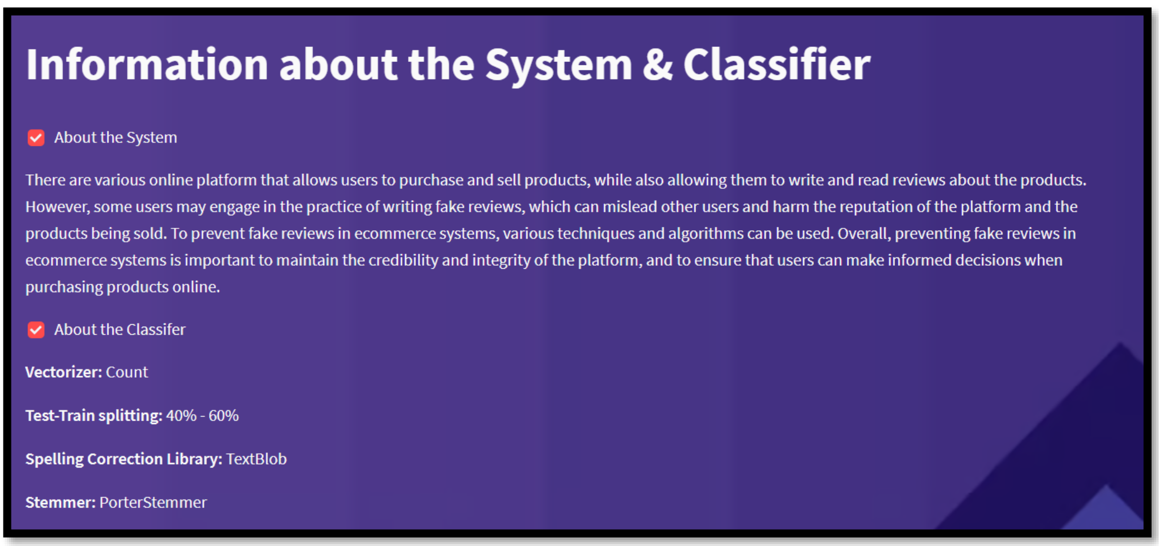
(1) User Interface:



Result 8.2.1: User Interface

Approaching the interface, the user interface that appears when accessing the website is shown in Result 8.2.1. As soon as a user inputs a review, the input is sent backend to determine whether the supplied words or sentences are legitimate or fraudulent, and accordingly, the detection is displayed on the interface.

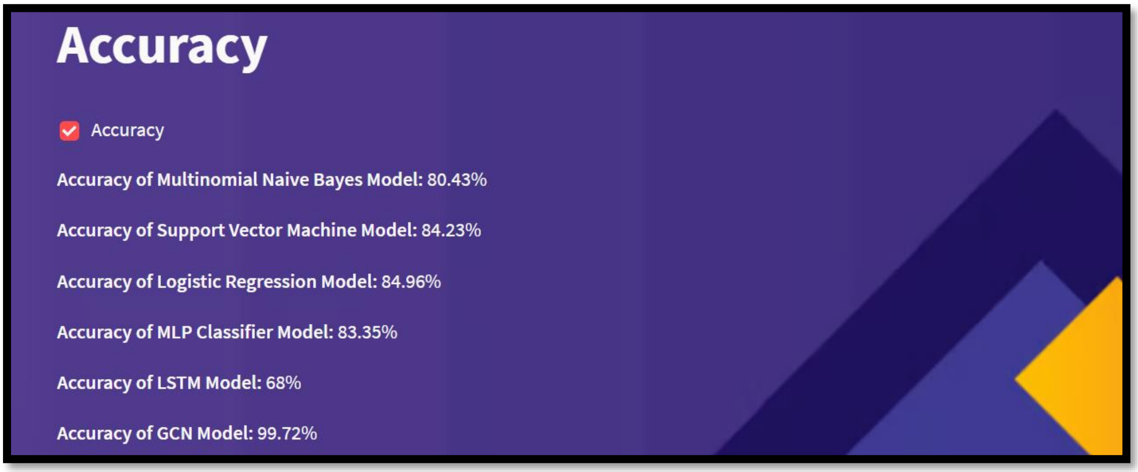
(2) Information about the System & Classifier:



Result 8.2.2: Information about the System & Classifier

Regarding Result 8.2.2, we have detailed our system and provided details about our classifier, such as the name of the vectorizer, the library we used for spelling correction, and the stemmer, we divide the test-train dataset into 40%–60%.

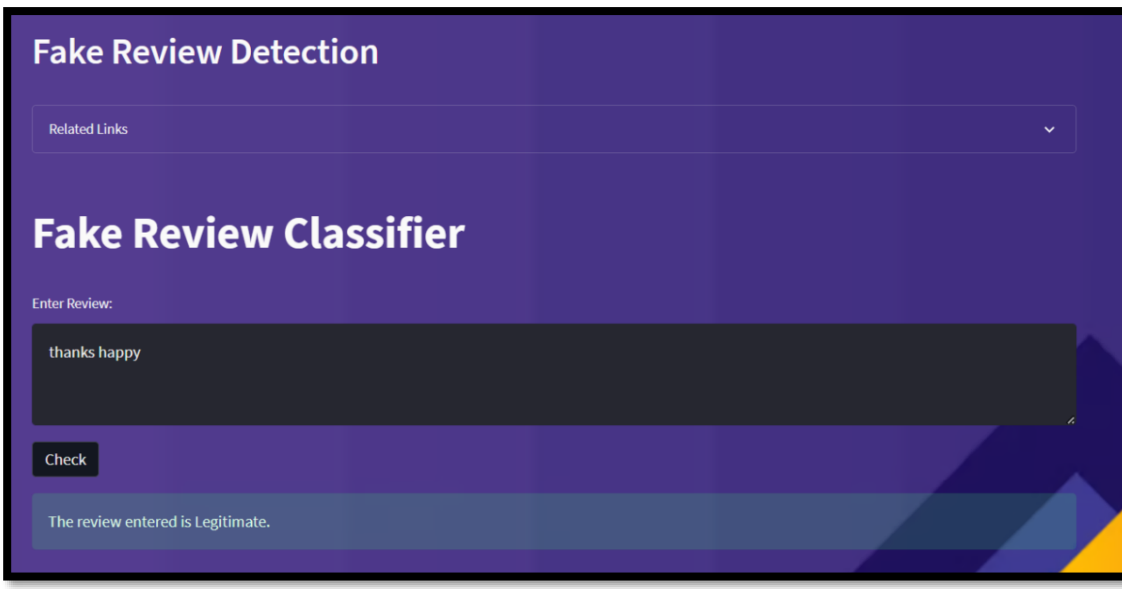
(3) Accuracy:



Result 8.2.3: Information about the System & Classifier

We deployed a menu of Accuracy on our website with a list of the algorithms we used, including Machine learning, Deep learning techniques, and primarily GCN, a subset of GNN, which provided the best accuracy rate among the rest; refer to the Result 8.2.3.

(4) Legitimate Review (Given review is from the dataset):



Result 8.2.4: Legitimate Review (Given review is from the dataset)

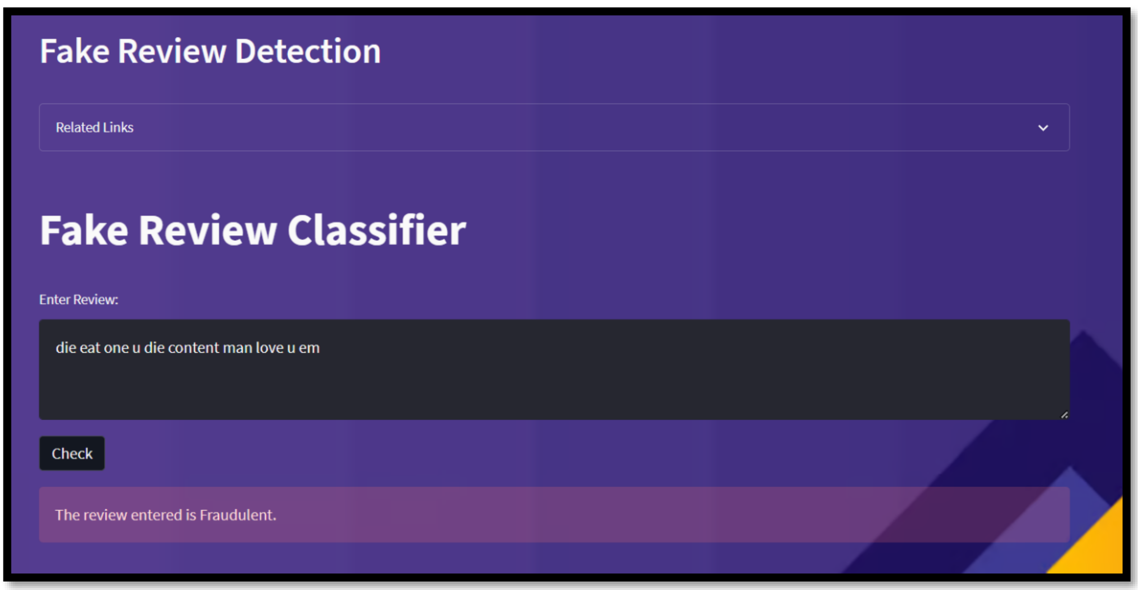
To determine whether the input in Result 8.2.4, is legitimate or fraudulent, we have chosen a sample review from the dataset with few words. We discovered that the input "thanks happy" was legitimate after going through each stage of the process, including exploring the data, preprocessing it, using NLP techniques like lemmatization, tokenization, and Porter stemming to train and test our model using machine learning, and deep learning algorithms.

(5) Legitimate Review (Review is not from the dataset):

Result 8.2.5: Legitimate Review (Review is not from the dataset)

Similarly, in Result 8.2.5, shown above, we choose a random sentence outside of the dataset with a limited number of words to test if the input is legitimate or not. We discovered the comment, "Awesome watch, it's too stylish," to be legitimate.

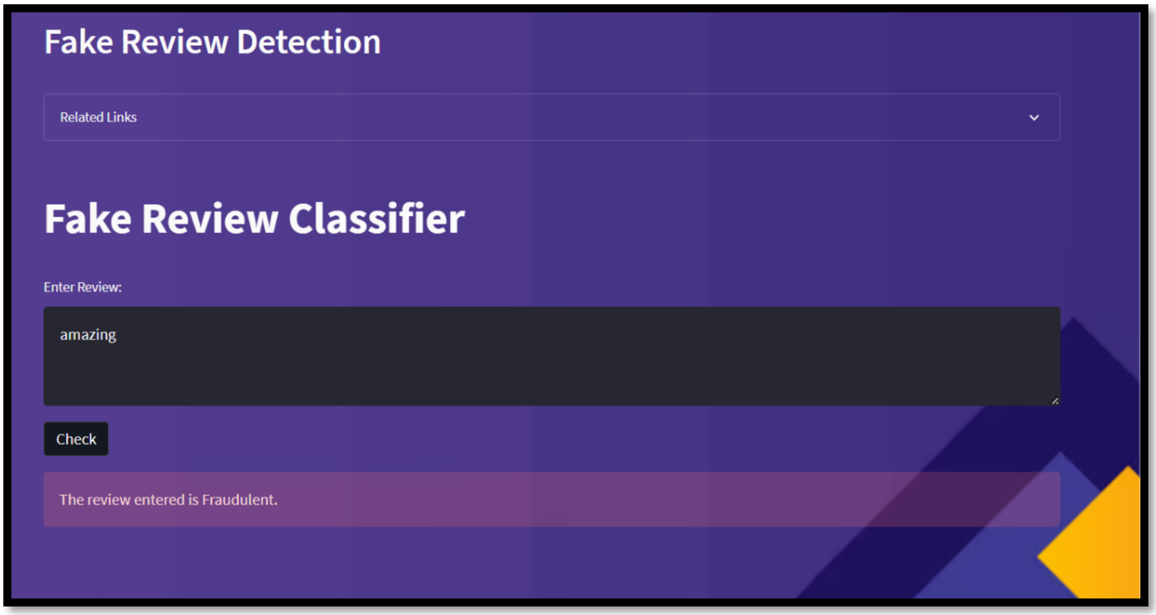
(6) Fraudulent Review (Given review is from the Dataset):



Result 8.2.6: Fraudulent Review (Given review is from the dataset)

Getting to the fraudulent aspect, in Result 8.2.6, we have taken input from the dataset, "die to eat one u die content man love u em," which demonstrated to be legitimate in the dataset but was predicted to be fraudulent after going through all the processes.

(7) Fraudulent Review (Review is not from the Dataset):



Result 8.2.7: Fraudulent Review (Review is not from the dataset)

Similarly, in Result 8.2.7, shown above, we choose a random word outside of the to test if the input is real or not. We discovered the comment, "amazing," to be fraudulent.

CHAPTER 9

Conclusion

The importance of reviews in influencing people's choices is highlighted by their significant effect on web-based data. The review plays a significant role in influencing consumer behavior, and they also have an impact on a variety of online company operations. Therefore, detecting fake reviews is an active and ongoing study field. This report presents a method for detecting fake reviews using machine learning and deep learning algorithms. The suggested method emphasizes the significance of both factors in identifying fake reviews by taking into account both reviewer conduct and review characteristics. The method provides a thorough and all-encompassing answer to the issue of identifying fake reviews in e-commerce applications by incorporating these two categories of features. The suggested method is assessed using the Kaggle dataset. To categorize which precision of the algorithms is the best, various machine learning and deep learning algorithms are used. Applying three feature extraction methods, including the Count Vectorizer, we implemented the fake review detection using a dataset. Three machine learning algorithms: Naive Bayes, Logistic Regression, SVM, and three deep learning algorithms: MLP Classifier, GCN, and LSTM are used to train and predict the extracted features. By utilizing the structural and semantic relationships between the words in the review corpus, these algorithms present a promising solution to this issue. We can conclude from the findings above that the GCN a GNN model provides good prediction accuracy.

CHAPTER 10

Future Scope

- To improve the quality of suggestions, extensive fieldwork could be done to obtain a better dataset with more accurate attributes.

- Global analysis of data from the online platform is possible thanks to the ease with which machine learning and deep learning algorithms can be scaled up or down to manage large volumes of reviews.

- Include multiple features. For instance, by identifying fake reviews, we can work to remove them from the platform. We can also identify fake goods and work to remove those from the platform.

References

[1] Zhao Li, Xin Shen, Yuhang Jiao, Xuming Pan, Pengcheng Zou, Xianling Meng, Chengwei Yao, and Jiajun Bu*et*, "Hierarchical bipartite graph neural networks: Towards large-scale E-commerce applications," in *Proceedings - International Conference on Data Engineering*, Apr. 2020, vol. 2020-April, pp. 1677–1688. doi: 10.1109/ICDE48307.2020.00149.

[2] Y. Wang, S. Qian, J. Hu, Q. Fang, and C. Xu, "Fake news detection via knowledge-driven multimodal graph convolutional networks," in *ICMR 2020 - Proceedings of the 2020 International Conference on Multimedia Retrieval*, Jun. 2020, pp. 540–547. doi: 10.1145/3372278.3390713.

[3] A. Mewada and R. K. Dewang, "Research on false review detection Methods: A state-of-the-art review," *Journal of King Saud University - Computer and Information Sciences*. King Saud bin Abdulaziz University, Oct. 01, 2021. doi: 10.1016/j.jksuci.2021.07.021.

[4] S. N. Alsubari, S. N. Deshmukh, M. H. Al-Adhaileh, F. W. Alsaade, and T. H. H. Aldhyani, "Development of Integrated Neural Network Model for Identification of Fake Reviews in E-Commerce Using Multidomain Datasets," *Appl Bionics Biomech*, vol. 2021, 2021, doi: 10.1155/2021/5522574.

[5] H. Paul and A. Nikolaev, "Fake review detection on online E-commerce platforms: a systematic literature review," *Data Min Knowl Discov*, vol. 35, no. 5, pp. 1830–1881, Sep. 2021, doi: 10.1007/s10618-021-00772-6.

[6] H. Yuan, J. Zheng, Q. Ye, Y. Qian, and Y. Zhang, "Improving fake news detection with a domain-adversarial and graph-attention neural network," *Decis Support Syst*, vol. 151, Dec. 2021, doi: 10.1016/j.dss.2021.113633.

[7] L. He, X. Wang, H. Chen, and G. Xu, "Online Spam Review Detection: A Survey of Literature," *Human-Centric Intelligent Systems*, vol. 2, no. 1–2, pp. 14–30, Jun. 2022, doi: 10.1007/s44230-022-00001-3.

[8] J. Salminen, C. Kandpal, A. M. Kamel, S. gyo Jung, and B. J. Jansen, "Creating and detecting fake reviews of online products," *Journal of Retailing and Consumer Services*, vol. 64, Jan. 2022, doi: 10.1016/j.jretconser.2021.102771.

[9] Saleh Nagi Alsubari, Sachin N. Deshmukh, Ahmed Abdullah Alqarni, Nizar Alsharif, Theyazn H. H. Aldhyani, Fawaz Waselallah Alsaade, and Osamah I. Khalaf, "Data analytics for the identification of fake reviews using supervised learning," *Computers, Materials and Continua*, vol. 70, no. 2, pp. 3189–3204, 2022, doi: 10.32604/cmc.2022.019625.

[10] Liu W, Zhang Y, Wang J, He Y, Caverlee J, Chan P, Yeung D, and Heng P, "Item Relationship Graph Neural Networks for E-Commerce," *IEEE Trans Neural Netw Learn Syst*, vol. 33, no. 9, pp. 4785–4799, Sep. 2022, doi: 10.1109/TNNLS.2021.3060872.

[11] C. Feng, He Y, Wen S, Liu G, Wang L, Xu J, and Zheng B, "DC-GNN: Decoupled Graph Neural Networks for Improving and Accelerating Large-Scale E-commerce Retrieval," in *WWW 2022 - Companion Proceedings of the Web Conference 2022*, Apr. 2022, pp. 32–40. doi: 10.1145/3487553.3524203.

[12] Varlamis, I, Michail, D, Glykou, F., and Tsantilas, P, "A Survey on the Use of Graph Convolutional Networks for Combating Fake News", doi: 10.3390/fi14030070.

Publication

In regards to our publication, we've submitted a research paper to the 4th International Conference on Data Science and Applications (ICDSA 2023).

Fake Reviews Detection for Ecommerce Using Machine Learning and Deep Learning Algorithms

Akshen Dhami
Department of Computer Engineering
A. P. Shah Institute of Technology
Mumbai, India
18102032@apsit.edu.in

Amish Nandu
Department of Computer Engineering
A. P. Shah Institute of Technology
Mumbai, India
18102048@apsit.edu.in

Sumati Hans
Department of Computer Engineering
A. P. Shah Institute of Technology
Mumbai, India
18102028@apsit.edu.in

Yash Pol
Department of Computer Engineering
A. P. Shah Institute of Technology
Mumbai, India
19102068@apsit.edu.in

Prof. Bharti Khemani
Department of Computer Engineering
A. P. Shah Institute of Technology
Mumbai, India
bjkhemani@apsit.edu.in

Prof. Sachin H Malave
Department of Computer Engineering
A. P. Shah Institute of Technology
Mumbai, India
shmalave@apsit.edu.in

***Abstract*—Dishonest opinions, usually referred to as fake reviews, are utilized to misled individuals and have recently gained relevance. This is a result of the swift growth in internet marketing transactions, including buying and selling. Customers have the option to write reviews and comments on the product or service they have purchased using e-commerce. Before making a purchasing decision, new buyers frequently read the website's reviews or comments. How new people, however, can tell real evaluations from false ones, which later mislead customers, cause losses, and damage businesses' reputations, is now an issue. The goal of the current study is to create an intelligent system that can identify false reviews on e-commerce platforms utilizing the n-grams of the review text and the reviewer's sentiment scores. The proposed methodology used in this work employed a Count Vectorization and Term Frequency-Inverse Document Frequency (TF-IDF) approach for feature extraction and their representation, together with a standard fake ecommerce review dataset for experimentation and data pretreatment procedures. N-grams from review texts were entered into the built models for detection and classification to determine if they were fraudulent or real. However, a dataset obtained from the Amazon website was used to train and test three different supervised machine-learning techniques, naive Bayes (NB), support vector machine (SVM), logistic regression (LR) and three deep-learning techniques, MLP Classifier, Graph Convolutional Networks (GCN) and Long Short-Term Memory (LSTM). We construct a graph representation of the review corpus, where each node corresponds to a word in the review and edges represent the co-occurrence of words. The GCN is then used to learn a representation of the graph, which captures the structural and semantic relationships among the words. The LSTM is then used to capture the temporal dependencies of the reviews and to classify them as fake or genuine.**

Keywords—Fake reviews; reviews detection on social Web; machine learning; deep learning; data Science.

I. INTRODUCTION

With web and multimedia development, the way people acquire information has significantly changed. Nowadays, there are more and more people consuming reviews through e-commerce apps, which may also provide all kinds of multimedia information on events taking place all over the world. Unfortunately, e-commerce websites even have fostered various fake reviews which usually contain misrepresented or even forged multimedia content, to mislead the readers and obtain rapid spread. Some evil guys even use rumors to mislead popular opinion, which may damage the credibility of the government on purpose. Therefore, it's necessary and urgent to use an automatic detector to prevent fake reviews from causing serious negative effects and make users receive truthful information.[1]

Thus far, with various fake reviews detection approaches [12], [13], including both traditional learning and deep learning-based models, consecutive and long-range semantic relations among words in feature representation becomes more and more important for fake news detection. Have been used to debunk fake news and minimize its harmful effects. Earlier studies on fake reviews detection constructed features through heuristic rules or statistical information. For instance, SVMTS [12] utilizes heuristic rules and a linear SVM to classify fake reviews on Amazon and uses a time-series structure to model the social feature variations. With the good success of the neural network, existing deep learning models have achieved performance improvement over traditional ones thanks to their superior ability of feature extraction. Convolutional neural networks (CNNs) [13] are introduced to get high-level representations from the text content of the post to identify rumors. Recurrent neural networks (RNNs) are needed to learn the hidden representation and sequential features from the propagation of fake reviews.

However, most of the prevailing deep learning methods only capture local semantic features in small sliding windows (short messages or word-level syntactic) for fake reviews detection, and ignore the structural information of posts which may be a very important aspect of fake reviews detection. For instance, some posts may have many words, and understanding the semantics of them needs modeling non-consecutive phrases and long-range word dependency. For instance, the relatively long post text, during which the keywords are "David Morrison", "impeachment", “Jake Burmer" then on, the mixture of these keywords indicates that the semantics may be related to "David Morrison supported the impeachment of Jake Burmer". However, these keywords aren't grouped and distributed throughout the whole post text. It's hard to capture the dependency [14] of semantic and structural information in a small sliding window. Therefore, the way to effectively capture the non-consecutive and long-range semantic relations among words in feature representation becomes more and more important for fake reviews detection.

Furthermore, fake reviews detection is sort of different from other classification tasks such as text classification, because, in most situations, it must detect fake reviews from various fields that the model may never have seen. However, existing deep learning methods typically specialize in inferring clues from the post-text content, and think little of the visual information and background of posts which humans also use in judging the credibility of an event. For instance, to gauge the credibility of the post, the primary thing people usually do is to observe the picture and then read the text content, and realize that Michael Bloomberg is an American politician whose Democratic Party is one of the two major political parties in modern America and he supports the impeachment against Donald Trump, and eventually give judgment. This means that social media posts carry a great deal of latent knowledge level connections and multimodal property, which may help us to judge whether the post is fake [1]. Hence, the way to acquire the background knowledge of the post text content, and fuse the textual information, knowledge concepts, and visual information of the post during a principled way is the key to fake news detection.

In recent years, Graph Neural Networks (GNNs) have shown remarkable performance in various applications, including natural language processing. GNNs can efficiently capture the complex relationships among the data points, represented as nodes and edges in a graph structure. These relationships can be structural, semantic, or temporal, and GNNs can effectively learn and propagate them to neighboring nodes to make accurate predictions.

In this paper, we propose a novel approach for fake review detection in e-commerce applications using GNN algorithms like Graph Convolutional Networks (GCN) and Long Short-Term Memory (LSTM). The proposed approach takes advantage of the graph structure of the review corpus to learn a representation that captures the underlying semantics and relationships among the words in the review. The GCN algorithm is used to learn a representation of the graph, which captures the structural and semantic relationships among the words. The LSTM is then used to capture the temporal dependencies of the reviews and to classify them as fake or genuine.

II. EXISTING SYSTEMS

A. Literature Survey

[1] The majority of them disregard the background information that is concealed in the post's text and makes it easier to spot bogus news. In order to model the semantic representations, a unique Knowledge-driven Multimodal Graph Convolutional Network (KMGCN) was developed. It unifies textual information, knowledge ideas, and visual information into a single framework for false news detection. Transformed text content into a graph instead of perceiving it as word sequences. To extract the semantic representation of these graphs, a well-designed graph convolutional network was used.

[2] In the year 2020, HiGNN was performed to obtain hierarchical user and item embeddings simultaneously and accurately forecast user preferences on a bigger scale by stacking multiple GNN modules and employing a deterministic clustering algorithm alternately. HiGNN (Hierarchical bipartite Graph Neural Network) obtained the hierarchical structure effectively and efficiently and produced a notable improvement when compared to state-of-the-art baselines in Taobao (one of the biggest real world e-commerce platforms).

[3] In the year 2021, Research on false review detection Methods: A state-of-the-art review involves summarizing the key issues of the current false review text detection research from the standpoint of the technical route of spam review detection research.

[4] “Development of Integrated Neural Network Model for Identification of Fake Reviews in ECommerce Using Multi Domain Datasets” published in the year 2021, the LSTM layer, which is based on gate mechanisms, is paired with the CNN technique for learning and processing the contextual information of the text's n-gram characteristics. The final layer of the proposed model, a sigmoid activation function, conducts a binary classification task of reviewing text into fake or true, using input sequences from the previous layer as input.

[5] In the year 2021, Fake review detection on online E-commerce platforms: a systematic literature review (Himanshu Paul, Alexander Nikolaev), research summarize the in order to confuse and mislead customers, malicious actors—review spammers—create and spread opinion-based false material that is referred to as fake reviews. Automatically, all paid reviews are fraudulent. For instance, even well-known online retailers encourage their consumers to post evaluations for recently acquired products; these reviews may be identified as having been "collected as part of a promotion."

[6] Improving fake news detection with domain-adversarial and graph-attention neural network papers was published in the year 2021. Where they used DAGA-NN. The results of the experiment on two genuine corpora demonstrated that DAGA-NN outperformed cutting-edge baselines.

[7] In the year 2022 Online Spam Review Detection: A Survey of Literature, More and more spam reviews were being added to online platforms in order to harm product reputations and merchant earnings. Positive, negative, or neutral evaluations of this nature all share the trait of misinforming customers or tarnishing reputations. Many researchers have been studying the detection of spam reviews using statistical or deep learning techniques with diverse datasets over the last ten years. Then, we thoroughly wrap up the currently used methodology and the datasets that are accessible. Third, we list the current network-based strategies for handling this task and make some recommendations for further research.

[8] In the Year 2022, Creating and detecting fake reviews of online products: Proves that machine learning classifiers are far much better in this area then other, recognizing evaluations produced by other machines with virtually flawless accuracy. This suggests that in the fight against bogus reviews, "machines can fight machines." The dataset used had a total of around 400 written fake reviews by crowd workers recruited via Amazon Mechanical Turk.

[9] In the year 2022 Data analytics for the identification of fake reviews using gold standard dataset developed by Ott et al involves applying supervised machine-learning algorithms, including naive Bayes, support vector machines, random forests, and adaptive boost, were investigated and put

into practice for the purpose of identifying false reviews. Utilizing the TF-IDF approach, features were extracted.

[10] In the year 2022, Item Relationship Graph Neural Networks for E-Commerce, IRGNN, a GNN-based framework was implemented. Take into account the data from items that are directly connected. In reality, the connectedness of goods located a few hops apart also has rich semantics that can be used to enhance relationship prediction. In the current study, we structure the problem as a multilabel link prediction task and suggest a novel graph neural network-based framework, item connection graph neural network (IRGNN), for simultaneously identifying numerous complex links. By recursively updating node embeddings with the messages from their neighbors, we add multihop relationships of products.

[11] In the year 2022, DC-GNN: Decoupled Graph Neural Networks for Improving and Accelerating Large-Scale E-commerce Retrieval. Pre-train, deep aggregation, and CTR prediction are the three stages where DC-GNN decouples the conventional framework. The training efficiency can be significantly increased using DC-GNN by separating the graph operations and the CTR prediction. In order to improve model performance, it can also enable deeper graph operations to properly explore higher-order proximity. Numerous studies using sizable industrial datasets have shown that DC-GNN significantly improves model performance and training effectiveness.

B. *Limitations of Existing Systems*

The prime factor to be considered is the kind of data used and the appropriate labeling techniques. It is important to keep in mind that the datasets for the analysis of the veracity of reviews on various E-commerce websites have recently been developed.[9] Earlier there were no smartphones available to each and every person so people were not able to write the reviews about the product which they are buying through these websites. Also, the websites creators themselves write reviews about a particular product, to gain customers' attention so that the customers buy their products. The buyers don't know which are the real reviews and which are the fake ones. Also, the drawback of e-commerce is the issue of security. People fear to provide personal and financial information, even though several improvements have been made in relation to data encryption. Certain websites do not have capabilities to conduct authentic transactions. Fear of providing credit card information and risk of identity limit the growth of e-commerce.[1] Many websites do not have high encryption for secure online transactions or to protect online identity. Some websites illegally collect statistics on consumers without their permission. Lack of privacy discourages people from using the internet for conducting commercial transactions. People have to rely on electronic images to purchase products. Sometimes, when the products are delivered, the product may not match with electronic images. Finally, it may not suit the needs of the buyers. The lack of 'touch and feel' prevents people from online shopping. The cyber laws that govern e-commerce transactions are not very clear and vary from country to country. These legal issues prevent people from entering into electronic contracts. Previously there were no such systems available to detect the reviews which are the real ones and which are the fake ones so that people rely on that and purchase products from those websites. A deep learning framework, which uses neural networks and the short-term memory architecture was adopted by many of the proposed models. The purpose of this neural network model is to enhance the detection of false reviews. The idea of stop words has been used for data preprocessing before training the model, which improves the model's accuracy. To construct an automated system for e-commerce websites, where fake news identification has become equally important, the future direction is in improvising and stretching the present work.[5]

III. Proposed System

In this section, the proposed work is demonstrated in detail. The steps which are involved in the proposed work are shown in Figure.1 The dataset used in this study is collected from various social sites like Amazon

- Data Initialization: We are initializing the data i.e., taking the raw data from the Kaggle dataset for further processing and algorithms for classification.
- Data Scraping: To make new and efficient data which contains useful information we need like the no. Of real and fake reviews.
- Data Collection: We obtained information on Amazon's online store through Kaggle, and we later processed the information in our notebook.
- Data Cleaning/Exploration: Data cleaning to clean the data i.e., to remove null values and unwanted rows or columns from the dataset. Here we are exploring the dataset to check number of unwanted columns, no. of null values and stop inwards as a part of feature extraction from dataset

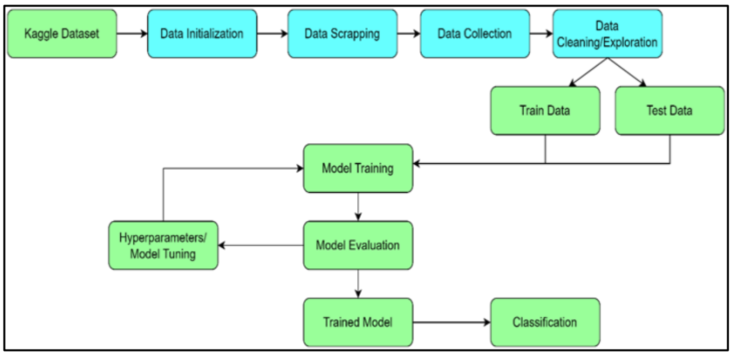


Figure.1: Architecture Diagram

A. *Architecture Diagram*

First, from the evaluations we have gathered from Kaggle, we have gathered both valid and non-legitimate data. However, there are some ways and procedures to determine which data is legitimate and which is not. We train and test the data after preprocessing, and classification follows next. Finding a function to divide the dataset into classes based on several parameters is the process of classification. In classification, data is divided into various classes by a computer program that has been trained on the training dataset.

B. *Flow Diagram*

A flow diagram is a visual representation of how data "flows" through an information system, simulating certain features of its operation. It is frequently a first step taken to develop a system overview that can then be developed. Figure. 2 shows the processing rate at which our project is progressing. As shown in the picture, we initially acquired or stored the data on our database after collecting it from the Kaggle dataset. The next step is data preprocessing, which employs strategies like data cleansing and data exploration to hunt for unused rows and columns in the dataset. Following the cleansing of the raw data, we trained and tested the data by training our model to determine the highest level of accuracy. Following model training, we assess the model and simultaneously fine-tune it to produce the desired model, which is then followed by the final flow, or classification.

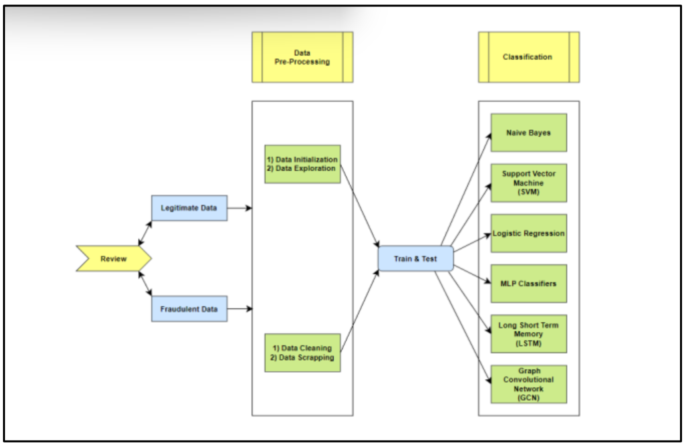


Figure. 2: Flow Diagram

IV. Methodology

The classification methodology is presented in this section. In this approach, the dataset is classified using machine learning and deep learning. The dataset collecting phase is the first step in this classification challenge, followed by executing the training and testing of the dataset after pre-processing, initialization, and finally, the suggested system is described by running the classifiers. Methodology. The algorithm is explained in detail below:

A. Naïve Bayes Algorithm:

The supervised machine learning technique Naive Bayes (NB) is utilized for classification. It is possible to use it to determine the likelihood of an occurrence given the likelihood of an earlier event. [9]

One of the well-known classification machine learning methods, the Naive Bayes Algorithm helps to categorize the data based on the computation of conditional probability values. It uses class levels represented as feature values or vectors of predictors for classification and applies the Bayes theorem to the computation. A quick algorithm for categorization issues is the Naive Bayes algorithm. Real-time prediction, multi-class prediction, recommendation systems, text categorization, and sentiment analysis use cases can all benefit from this technique. Gaussian, Multinomial, and Bernoulli distributions can be used to create the Naive Bayes algorithm. For a large data set, this approach is simple to use and scalable. It helps to calculate the posterior probability P(c|x) using the prior probability of class P(c), the prior probability of predictor P(x), and the probability of predictor given class, also called as likelihood P(x|c).

The formula or equation to calculate posterior probability is:

$$P(c|x) = \frac{P(X|C)\,P(c)}{P(x)} \tag{1}$$

Working:

An effective algorithm for issues involving numerous classes and text data analysis is naive bayes. Given that the Naive Bayes theorem is built on the Bayes theorem, it is crucial to first comprehend how the latter works.

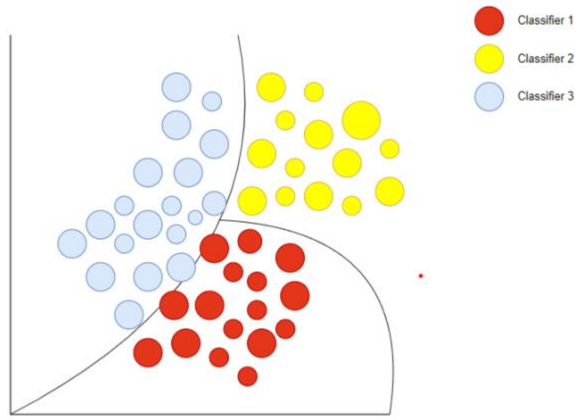


Figure. 3: Working of Naïve Bayes Classifier

Thomas Bayes developed the Bayes theorem, which determines the likelihood of an event happening based on knowledge of the circumstances surrounding the event. It is based on the following formula:

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)} \tag{2}$$

Where we are calculating the probability of class A when predictor B is already provided.

P(B) = prior probability of B

P(A) = prior probability of class A

P(B|A) = occurrence of predictor B given class A probability

This formula helps us to calculate the probability of the tags in the text. Let's use an illustration to better grasp the Naive Bayes method. We have included a data set of weather conditions that includes sunny, overcast, and wet days in the table below. Now we must forecast the likelihood that the players will really play based on the weather.

Accuracy: 80%

B. Support Vector Machine Algorithm:

SVM is used to categorize texts and performs well in high-dimensional vector spaces. Support Vector Machine (SVM) is a supervised machine learning technique which is used for both regression as well as classification. Despite the fact that we also refer to regression issues, categorization is the most appropriate term. Finding an N-dimensional space hyperplane that clearly classifies the data points is the goal of the SVM method. The number of features it generates affects the hyperplane's dimension. A line is all that the hyperplane is if there are only two input features. The hyperplane changes into a 2-D plane if the input feature count reaches three. When there are more than three aspects, it is challenging to imagine. The SVM kernel is defined as the function that converts a non-separable problem into separable problem by taking low dimensional input space and then transforming it into higher-dimensional space. . It works best in non-linear separation issues. Simply explained, the kernel determines how to split the data depending on the labels or outputs defined after performing some incredibly sophisticated data transformations. [9]

Working:

By concentrating on its main type, the SVM classifier, it is easiest to comprehend the SVM algorithm. The goal of the SVM classifier is to create a hyper-lane that divides the data points into several classes in an N-dimensional space. However, this hyper-pane is selected based on margin, taking into account the hyperplane offering the greatest margin between the two classes. The Support Vectors, a type of data point, are used to determine these margins. The data points that are close to the hyper-plane and aid in its orientation are known as support vectors.

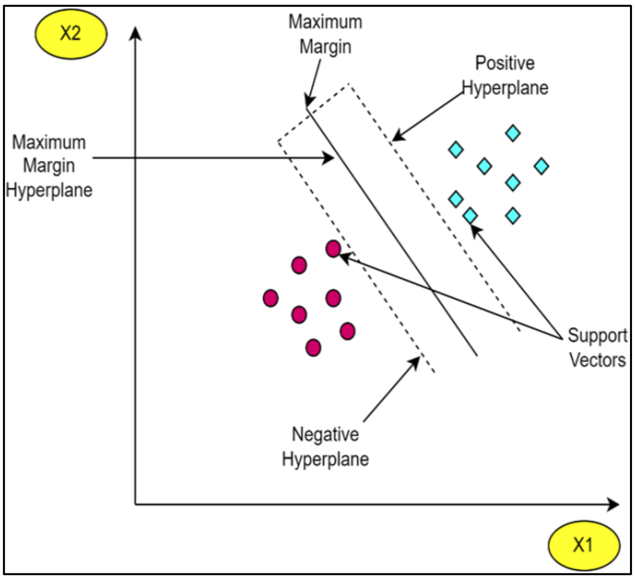


Figure. 4: Working of Support Vector Machine

Following are some mathematical methods in which the SVM classifier's operation can be understood:

First, the classes are predicted by the SVM method. The classes are given a number 1 for one and a number -1 for the other.

The second step is to transform the business issue into a mathematical equation with unknown variables, as is done by all machine learning algorithms. By turning the issue into an optimization issue, these unknowns are then discovered. In the case of the SVM classifier, a loss function called the hinge loss function is utilized and adjusted to find the maximum margin because optimization issues always aim at maximizing or decreasing something while searching for and adjusting for the unknowns.

High Loss Function

Third is for simplicity, this loss function can also be referred to as a cost function, whose cost is 0 when no class is predicted wrongly. If this is not the case, however, error/loss is computed. The issue with the present situation is that there is a trade-off between maximizing margin and the loss that is produced if the margin is maximized to an extreme degree. A regularization parameter is provided to theoretically connect these ideas.

$$c(x, y, f(x)) = \begin{cases} 0, \\ 1 - y * f(x) \end{cases} if\ y * f(x) \geq 1 \quad (3)$$

Loss Function for SVM

$$min \leftthreetimes ||w||^2 + \sum_{i=1}^{n} (1 - y_i < x_i, \text{w} >) \quad (4)$$

Fourth is with the majority of optimization problems, weights are optimized by computing gradients using partial derivatives, an intermediate calculus topic.

Gradients

$$\frac{\delta}{\delta w_k} \leftthreetimes \left||w|\right|^2 = 2 \leftthreetimes w_k \quad (5)$$

$$\frac{\delta}{\delta w_k}(1 - y_i < x_i, w >) = \begin{cases} 0, & if\ y_i < x_{i,}, w > \geq 1 \\ -y_i x_{ik}, & \\ & else \end{cases} \quad (6)$$

Fifth is that the gradients are updated only when there is no classification error using the regularization parameter, and when misclassification occurs using the loss function.

Updating of Gradients when there is No Misclassification

$$w = w - \alpha(2 \leftthreetimes w) \quad (7)$$

Updating of Gradients when there is Misclassification

$$w = w - \alpha(2 \leftthreetimes w)(y_i x_i - 2 \leftthreetimes w) \quad (8)$$

Sixth is that the gradients are only updated using the regularization parameter while there is no classification error, and when misclassification occurs, the loss function is also employed.

Accuracy: 84%

C. *Logistic Regression:*

LR exhibits the high accuracy and recall for the dataset even though the dataset was divided in such a way that the test is 85% and 82% is the training set.

The fundamentals of logistic regression are covered in this article, along with python's use of it. Essentially, supervised categorization is what logistic regression does. For a specific collection of features (or inputs), X, the target variable (or output), y, can only take discrete values in a classification issue. Despite what many people think, logistic regression IS a regression model. In order to determine the likelihood that a specific data entry falls under the category designated by the number "1," the program creates a regression model. Logistic regression models the data using the sigmoid function, just like linear regression assumes that the data follows a linear distribution.

Only when a decision threshold is included does logistic regression become a classification approach. The classification problem itself determines the threshold value, which is a crucial component of logistic regression. The precision and recall levels have a significant impact on the choice of the threshold value. In a perfect world, precision and recall should both equal 1, but this is rarely the case.

$$g(z) = \frac{1}{1+e^{-z}} \quad (9)$$

Working:

In Logistic Regression, the "sigmoid function" or "logistic function" is implemented as a cost function. Consequently, the sigmoid function can be used to forecast probability values.

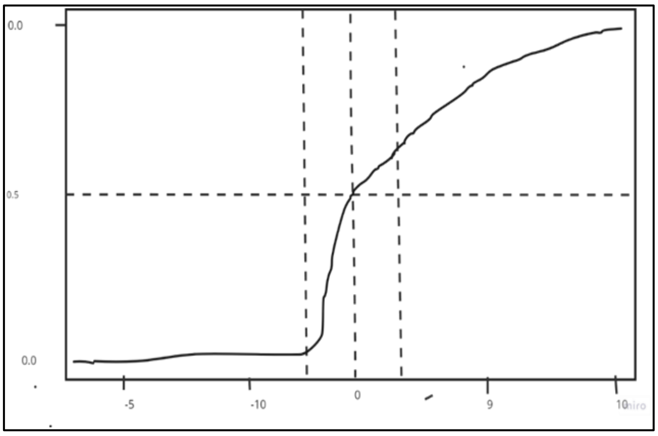


Figure. 5: Working of Logistic Regression

Sigmoid Function

The mathematical equation of Logistic Regression

First of all, let's have a look at the mathematical equation of the sigmoid function.

$$f(z) = \frac{1}{1-e^{-z}} \tag{10}$$

Now, in the above equation,

$$z = w_0 + w_1x_1 + w_2x_2 + \cdots w_nx_n \tag{11}$$

In the equation above, w0, w1 w2..., wn, is used to represent the regression of the model's co-efficient, which is determined from maximum likelihood estimation, and x0, x1 x2..., xn, is used to represent the features or independent variables. The binary outcome probability is calculated in the aforementioned equation by F (z), where the probabilities are divided into two groups based on the provided data point (x).

Accuracy: 85%

D. MLP Classifier:

A multi-layered perceptron (MLP) is one of the most commonly used neural network models used in the field of deep learning.

There can be more than one linear layer available in the multilayer perceptron (combinations of neurons). If we use the straightforward example of a three-layer network, the input layer will be on the top, the output layer will be at the bottom, and the middle layer would be referred to as the hidden layer. There is an input layer that receives our input data, and the output layer which receives our output. To make the model more complex in accordance with our purpose, we are free to increase the number of hidden layers as much as we like. The most basic type of artificial neural network is one that uses MLPs. They transform a set of inputs into a single output between 0 and 1 using a succession of perceptron's, or equations with inputs, outputs, and weights. The process continues until a single output is achieved by feeding that output into a subsequent layer of perceptron (or set of outputs, depending on the function of the MLP). The minimum number of layers in an MLP is three: an input layer, at least one hidden layer, and an output layer. It is simpler to identify data that cannot be linearly categorized thanks to the hidden layer. The loss function is established to gauge the classifier's effectiveness. If the anticipated class does not match the actual class, the loss will be considerable; otherwise, it will be minimal. When the model is being trained, the issue of overfitting and underfitting can occasionally arise. Our model performs admirably on training data in this instance, but not on testing data. An optimization technique is needed in order to train the network, and for this, a loss function and an optimizer are necessary. The values for the set of weights, W, that minimizes the loss function will be discovered using this technique.

Training the Model:

There are basically three steps in the training of the model.

1. Forward pass
2. Calculate error or loss
3. Backward pass

Working:

The following is the MLP learning process:

- Propagate data forward to the output layer starting with the input layer. The forward propagation stage is this one.
- Determine the mistake based on the results (the difference between the predicted and known outcome). The mistake must be kept to a minimum.
- Reverse the mistake. Update the model by determining its derivative with respect to each network weight.
- To learn the optimum weights, repeat the three processes listed above across a number of epochs.
- Finally, a threshold function is used to extract the data and produce the expected class labels.
- Forward Propagation in MLP In the first step, calculate the activation unit al(h) of the hidden layer.

$$z_1^{(h)} = a_0^{(in)}w_{0,1}^{(h)} + a_1^{(in)}w_{1,1}^{(h)} + \cdots + a_m^{(in)}w_{m,1}^{(h)} \tag{12}$$

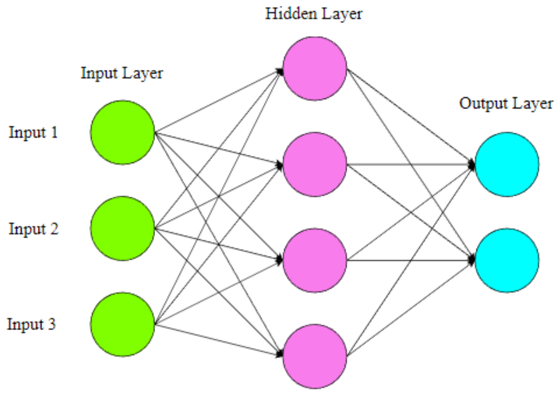


Figure. 6: Working of MLP Classifier

Applying an activation function to the z value yields an activation unit. To be able to learn weights using gradient descent, it must be differentiable. The sigmoid (logistic) function is frequently the activation function.

It enables the nonlinearity required to address challenging issues like image processing.

$$\emptyset(z) = \frac{1}{1+e^{-z}} \tag{13}$$

Accuracy: 84%.

E. Long Short-Term Memory (LSTM):

In the methodology is to use the LSTM algorithm to capture the temporal dependencies of the reviews and classify them as genuine or fake. The LSTM takes the learned representation of the graph as input and processes it sequentially, considering the temporal order of the reviews.
Let x(t) be the input feature vector at time t, h(t) be the hidden state at time t, and c(t) be the cell state at time t. The LSTM algorithm can be represented as follows:

$$i_t = \sigma_g(w_i \times x_t + U_i \times h_{t-1} + b_i) \quad (14)$$
$$f_t = \sigma_g\left(w_f \times x_t + U_f \times h_{t-1} + b_f\right) \quad (15)$$
$$o_t = \sigma_g(w_0 \times x_t + U_0 \times h_{t-1} + b_0) \quad (16)$$

where Wi, Wf, Wo, Wc are the input weight matrices, Ui, Uf, Uo, Uc are the recurrent weight matrices, bi, bf, bo, bc are the bias vectors, σ is the sigmoid activation function
The LSTM algorithm updates the hidden state and cell state at each time step by selectively forgetting and remembering information from previous time steps and integrating new information from the input.

Accuracy: 67%.

F. GCN:
The fifth step in the methodology is to use the GCN algorithm to learn a representation of the graph that captures the underlying semantics and relationships among the words. The GCN takes the graph representation of the review corpus as input and iteratively updates the node representations based on the features of neighboring nodes.
Let H(l) be the hidden representation of the graph at layer l, where H(0) is the input feature matrix, and H(l+1) is the hidden representation after layer l. The GCN algorithm can be represented as follows:

$$H^{(l+1)} = \sigma(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}) \quad (17)$$

where D is the degree matrix of the graph, W(l) is the weight matrix at layer l, and σ is the activation function. The GCN algorithm updates the node representations at each layer by aggregating the features of neighboring nodes and applying a linear transformation.

Accuracy: 99.7%.

V. Result and Conclusion

A. Outcome

We used the Kaggle-downloaded Amazon review datasets and subjected them to a number of machines learning and deep learning models. We employed the Multinomial Naive Bayes, Support Vector Machine, and Logistic Regression algorithms for machine learning algorithms, and deep learning algorithms namely: MLP Classifier, GCN and LSTM methods.

Our goal is to determine whether they are genuine or false. We utilized a Machine Learning & Deep Neural Network Algorithm to assess the accuracy of each approach separately & for improved results. Given the significant improvement in neural network research, we used a range of deep neural network models, which are addressed in more detail in following parts. We discuss many neural network topologies that are often used in research and go through numerous iterations of them. There is a high-level description and a list of how neural network topology's function.

An arrangement of nodes connected by edges, some of which may or may not be directed, is known as a graph. Each node has a certain collection of features, and the edges between them explain the connections between them. In a typical GNN, messages are passed between nearby nodes via the Edges. It makes sense that the neural network would encode the data that is transmitted from one node to its connected neighbors in the message. The representation of a node in a GNN is made up of the messages that have been accumulated from all of a node's neighbors to the current node at any layer. One can acquire a vector representation for each node after a number of message transmission iterations. This vector representation can be viewed as an embedding representation that describes the neighborhood network structure and the node's feature information. Social networks, images, chemical compositions, human brain neurons, and even a conventional, fully linked neural network can all be represented as graphs. Because of this, GNNs are quite useful. The following table & graphs shows the results arrive from our implementation model.

Naïve Bayes	80%
SVM	84%
Logistic Regression	85%
MLP Classifier	84%
LSTM	67%
GCN	99.7%

Table.1

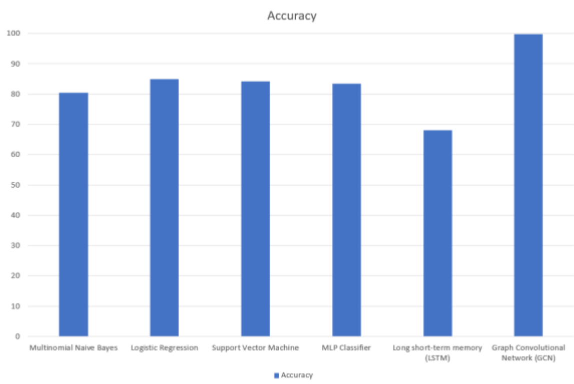


Figure.7

B. Conclusion

This paper highlights the significant impact of reviews on web-based data and emphasizes their importance in influencing people's decisions. It is evident that reviews have a crucial role to play in shaping consumer behavior, and their influence extends to various aspects of online business operations. Thus, fake reviews detection is a vivid and ongoing research area. In this paper, a machine learning fake reviews detection approach is presented. The proposed approach considers not only the characteristics of reviews but also the behavioral features of reviewers, emphasizing the importance of both aspects in the detection of fake reviews. By incorporating these two types of features, the approach offers a comprehensive and holistic solution to the problem of detecting fake reviews in e-commerce applications. The Kaggle dataset is used to evaluate the proposed approach. Different machine learning & deep learning algorithms are used to classify which accuracy of the algorithms is the best. We have implemented Fake review detection taken dataset by applying three feature extraction techniques namely Count Vectorizer, Engram model, TFIDF Vectorizer. The extracted features are trained and predicted using four machine learning algorithms namely Naïve Bayes, Logistic Regression, SVM and three deep learning algorithms namely: MLP Classifier, GCN and LSTM which offers a promising solution to this problem by leveraging the structural and semantic relationships among the words in the review corpus. From the above results we can understand that the Logistic Regression model gives good accuracy on prediction.

REFERENCES

[1] Y. Wang, S. Qian, J. Hu, Q. Fang, and C. Xu, "Fake news detection via knowledge-driven multimodal graph convolutional networks," in *ICMR 2020 - Proceedings of the 2020 International Conference on Multimedia Retrieval*, Jun. 2020, pp. 540–547. doi: 10.1145/3372278.3390713.

[2] Z. Li *et al.*, "Hierarchical bipartite graph neural networks: Towards large-scale E-commerce applications," in *Proceedings - International Conference on Data Engineering*, Apr. 2020, vol. 2020-April, pp. 1677–1688. doi: 10.1109/ICDE48307.2020.00149.

[3] A. Mewada and R. K. Dewang, "Research on false review detection Methods: A state-of-the-art review," *Journal of King Saud University - Computer and Information Sciences*. King Saud bin Abdulaziz University, Oct. 01, 2021. doi: 10.1016/j.jksuci.2021.07.021.

[4] S. N. Alsubari, S. N. Deshmukh, M. H. Al-Adhaileh, F. W. Alsaade, and T. H. H. Aldhyani, "Development of Integrated Neural Network Model for Identification of Fake Reviews in E-Commerce Using Multidomain Datasets," *Appl Bionics Biomech*, vol. 2021, 2021, doi: 10.1155/2021/5522574.

[5] H. Paul and A. Nikolaev, "Fake review detection on online E-commerce platforms: a systematic literature review," *Data Min Knowl Discov*, vol. 35, no. 5, pp. 1830–1881, Sep. 2021, doi: 10.1007/s10618-021-00772-6.

[6] H. Yuan, J. Zheng, Q. Ye, Y. Qian, and Y. Zhang, "Improving fake news detection with domain-adversarial and graph-attention neural network," *Decis Support Syst*, vol. 151, Dec. 2021, doi: 10.1016/j.dss.2021.113633.

[7] L. He, X. Wang, H. Chen, and G. Xu, "Online Spam Review Detection: A Survey of Literature," *Human-Centric Intelligent Systems*, vol. 2, no. 1–2, pp. 14–30, Jun. 2022, doi: 10.1007/s44230-022-00001-3.

[8] J. Salminen, C. Kandpal, A. M. Kamel, S. gyo Jung, and B. J. Jansen, "Creating and detecting fake reviews of online products," *Journal of Retailing and Consumer Services*, vol. 64, Jan. 2022, doi: 10.1016/j.jretconser.2021.102771.

[9] S. N. Alsubari *et al.*, "Data analytics for the identification of fake reviews using supervised learning," *Computers, Materials and Continua*, vol. 70, no. 2, pp. 3189–3204, 2022, doi: 10.32604/cmc.2022.019625.

[10] W. Liu *et al.*, "Item Relationship Graph Neural Networks for E-Commerce," *IEEE Trans Neural Netw Learn Syst*, vol. 33, no. 9, pp. 4785–4799, Sep. 2022, doi: 10.1109/TNNLS.2021.3060872.

[11] C. Feng *et al.*, "DC-GNN: Decoupled Graph Neural Networks for Improving and Accelerating Large-Scale E-commerce Retrieval," in *WWW 2022 - Companion Proceedings of the Web Conference 2022*, Apr. 2022, pp. 32–40. doi: 10.1145/3487553.3524203.

[12] W. Gao *et al.*, "Detecting rumors from microblogs with recurrent neural networks," 2016. [Online]. Available: https://ink.library.smu.edu.sg/sis_research

[13] F. Yu, Q. Liu, S. Wu, L. Wang, and T. Tan, "A Convolutional Approach for Misinformation Identification," 2017. [Online]. Available: http://www.npr.org/2016/11/08/500686320/did-social-media-

[14] H. Peng *et al.*, "Large-scale hierarchical text classification with recursively regularized deep graph-CNN," in *The Web Conference 2018 - Proceedings of the World Wide Web Conference, WWW 2018*, Apr. 2018, pp. 1063–1072. doi: 10.1145/3178876.3186005.