Fake News Detection using ML Classifier

Submitted in partial fulfillment of the requirements for the degree of

Bachelor of Technology
In
Computer Science and Engineering

by

KAARTIKEYA PANJWANI

19BCE2124

Under the guidance of

Prof. / Dr.

SHALINI L.

VIT, Vellore.

May, 2023

DECLARATION

I hereby declare that the thesis entitled “Fake News Detection using ML Classifier” submitted by me, for the award of the degree of *Bachelor of Technology in Computer Science and Engineering* to VIT is a record of bonafide work carried out by me under the supervision of Prof. Shalini L.

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place : Vellore

Date : 19/05/2023

Signature of the Candidate

CERTIFICATE

This is to certify that the thesis entitled "Fake News Detection using ML Classifier" submitted by Kaartikeya Panjwani **& 19BCE2124**, **School of Computer Science and Engineering (SCOPE)**, VIT, for the award of the degree of *Bachelor of Technology in Computer Science and Engineering*, is a record of bonafide work carried out by him / her under my supervision during the period, 01. 07. 2022 to 30.04.2023, as per the VIT code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfills the requirements and regulations of the University and in my opinion meets the necessary standards for submission.

Place : Vellore

Date : 19/05/2023

Signature of the Guide

Internal Examiner

External Examiner

Head of the Department Btech,
Computer Science and Engineering

ACKNOWLEDGEMENTS

I would like to express my special thanks to our project guide Prof. Shalini L for her time and efforts she provided throughout the year. Her constant guidance and support has kept us motivated to give our best to the project. This project is a high reflection of her ideas and suggestions. Her useful advice and suggestions were really helpful to us during the project's completion. In this aspect, we are eternally grateful to you.

I would also like to express our profound gratitude to Dr. VAIRAMUTHU S (HOD), of the department SCOPE, and Dr. RAMESH BABU K (Dean) of Vellore Institute of technology, Vellore for giving us the opportunity to work on this project and their contributions to the completion of my project titled Fake News Detection using ML Classifier.

I would like to acknowledge that this project was completed entirely by myself, Shashwat Gupta and Aparna Sahu and not by someone else.

Kaartikeya Panjwani

Executive Summary

Machine learning has been essential in the classification of the data, despite its limitations. This thesis investigates various machine learning methods for spotting fake and manufactured news. The Internet's rapid and widespread spread of bogus news emphasizes the need for automated hoax detection systems. Machine learning (ML) techniques can be applied in the context of social networks towards this objective. Traditional methods for identifying fake news rely on content analysis (i.e., studying the news's substance) or, more recently, social context models, such as mapping the news' diffusion pattern. The number of people using the internet has increased dramatically in recent years, and the number of people using the internet at home has increased even more. When political polarization and public mistrust of their leaders rise, the dissemination of this kind of misinformation poses a serious threat to social cohesion and wellbeing. Hence, the epidemic of fake news is having a big impact on our social lives, especially in politics. In order to address this issue, this study suggests brand-new methods for the fake news detection system based on machine learning (ML) and deep learning (DL). Finding the best model with good accuracy performance is the primary goal of this thesis. We propose a Machine Learning classifier strategy to identify if news items and posts are genuine or not because human filtration of news articles and posts on a global scale is an impractical approach. In this thesis, we first look into a number of artificial intelligence classifiers and compare them for accuracy. Then, for constructing our fake news detector, which accepts the article and its author as input and outputs whether the piece is fake or not, we employ the most accurate classifier.

CONTENTS

Page No.

List of Figures

List of Tables

List of Abbreviations

DL	Deep learning
ML	Machine learning
NLP	Natural Language Processing
IR	Information Retrieval
FE	Feature Extraction
DM	Data Mining
SNA	Social Network Analysis
SVM	Support Vector Machine
LSTM	Long Short Term Memory
KNN	K-Nearest-Neighbours
TFIDF	Term Frequency Inverse Document Frequency
NB	Naïve Bayes
LR	Logistic Regression
RF	Random Forest
DNN	Deep Neural Network
TF	Term Frequency
IDF	Inverse Document Frequency
MTBF	Mean Time Between Failures
AWS	Amazon Web

1. Introduction

1.1 Theoretical Background

Fake news is a big problem, it has made a huge impact in the discourse and even been used to manipulate mass human behavior. Information is power and power in the wrong hands can always be very dangerous. We need systems that can verify information thus only the truth can be spread out to people or else at least we can inform the people that some information is much likely to be true. As fake news is generated by humans (by mistake or intentionally) it's very difficult to classify it. We need models that can understand human mind complexity and also have a vast knowledge-bank to check what information can be correct. Thus we need machine learning to understand the patterns of fake news and classify it. machine learning models take in vast amounts of data and analyze the patterns. We can analyze the patterns that we see in fake news. Logistic regression is a supervised machine learning algorithm mainly used for classification tasks where the goal is to predict the probability of an instance of belonging to a given class or not. Some key theoretical approaches for fake news detection are:

- Deep learning (DL) and machine learning (ML): These techniques are frequently used in the detection of fake news. To determine if news stories are authentic or not, supervised learning algorithms such as Naive Bayes, Support Vector Machines (SVM), decision trees, and neural networks are trained on labelled datasets.
- Natural Language Processing (NLP): NLP plays a fundamental role in fake news detection. Techniques such as text preprocessing, tokenization, part-of-speech tagging, syntactic and semantic analysis, named entity recognition, and sentiment analysis are used to extract meaningful features from text data. NLP models and algorithms help in understanding the linguistic patterns, contextual cues, and semantic relationships within news articles.
- Information Retrieval (IR): In order to support the fake news identification process, IR techniques are employed to gather pertinent data and sources. These methods involve indexing and searching through enormous databases of news stories, using retrieval models, and sorting documents according to how pertinent they are to a particular query. Fact-checking, source reputation research, and the collecting of extra

information to support or refute statements made in news articles can all be aided by IR techniques.

- Data Mining (DM) and Feature Extraction (FE): Large datasets can be explored using data mining techniques to find significant patterns, trends, and relationships that can be used to spot fake news. Choosing and creating useful features that capture the traits that set fake news stories apart, such as lexical, syntactic, semantic, or stylistic elements, is the process of feature extraction. The effectiveness of false news detection models can also be increased by using feature selection techniques and dimensionality reduction strategies.
- Social Network Analysis (SNA): SNA looks at how different social network components relate to one another. Social network analysis (SNA) is useful for studying the dynamics of propagation, influential individuals, and patterns of information dissemination since fake news frequently circulates through social media platforms. In order to identify and stop the spread of fake news, it can be helpful to analyze the social network structure, user engagement patterns, and content sharing behaviors.

1.2 Motivation

In the era of news in our lives, in the ideal world people would have been responsible enough to not share any misleading information as there are many sources available now-a-days. Fake news such as spam messages, funding news or any false information should be prevented from reaching the masses. We consider it as a serious issue although it is extremely complicated to find out which is fraud and which is not in social media, they are replicated as the original one. As the technology evolved and the machine intelligence has come into existence everyone tends to use available sources for creating and dissemination of fraud news. People who are illiterate might be new to digital media as they are inexperienced, so they are the ones who believe that fraud news easily and makes it practical in their lives. The motivation behind this work is to help people improve the consumption of legitimate news while discarding misleading information in social media. Classification accuracy of fake news may be improved from the utilization of machine learning ensemble methods.

1.3 Aim of the Proposed Work

In recent years, India has seen the world's greatest rise in number of active internet users as a result of low-cost pricing of data plans and competitive environment amongst major network providers. Hence, as active internet users have grown, the amount of fake news through online articles and social media has also increased. The problem of false news on the internet is becoming more and more prominent as news articles are spread online. Popular social media sites like Facebook, Instagram, Twitter, etc. are used extensively to spread false or incorrect information in the form of videos, posts, articles, and URLs.

These fake news articles are mostly politically driven and are aimed at radicalizing and dividing a certain section of people. This radicalization can ultimately lead to nation-wide unrest and improper functioning of the society.

As a result, editors and journalists require new technologies that will speed up the process of verifying the information that has come via social media. With the expansion of artificial intelligence and machine learning in our daily lives, it is only befitting that we work towards an autonomous fake news detection system. Although there have been multiple approaches regarding detection of false news through machine learning, almost all either only focus on one topic (e.g. Sports or politics) or have low accuracies. Hence in this thesis, we aim to discover and apply a new approach to detect fake news through machine learning classifiers and pre-indexed data.

1.4 Objectives of the proposed work

Our approach involves utilizing machine learning classifiers to detect whether the news article provided as input by the user is fake or not. Hence, firstly we compare various machine learning classifiers, namely Logistic Regression, Naïve Bayes Classifier, K- Nearest Neighbors Classifier, Decision Tree Classifier, and Random Forest Classifier, on pre-indexed data to find out which one of them suits the best for our approach. Driven by the requirement for automated detection of false news, our objective is to determine which classification model identifies counterfeit features accurately.

We will be comparing these classifiers through the parameters like their speed, transparency, memory usage, and most importantly their accuracy. After deciding a classifier for our thesis, we proceed with the pre-processing of our data and applying the model using Python. We will be using the feature extraction technique, Term Frequency-Inverse Document Frequency (Tf-Idf) which will be utilized to convert the textual data into numerical data.

We start our training process by splitting the data into training and testing data and feeding the classifier pre-labelled data with true and false values against the news articles in the dataset. After training our machine learning model, we begin our prediction process which involves feeding unlabeled articles from our dataset to our model. The expected outcome is the label for the corresponding articles which will indicate whether the news articles are real or fake.

2. Literature Survey

2.1. Survey of the Existing Models/Work

The authors of [1] compared different ML models and classifiers and found out the highest f1 score and accuracy and of 97% with the linear regression model. They also summarized that for some datasets, the SVM model yielded the highest accuracy of 98%[1]. The authors concluded that the ensemble learners have shown an overall better score on all performance metrics as compared to the individual learners. However, the authors have missed on the opportunity to compare certain combinations of classifiers in ensemble learning. They have also not mentioned the key-phrases used to identify the validity of the articles and the genres of articles studied. Graph theory and machine learning techniques can be employed to identify the key sources involved in spread of fake news.[1] Likewise, real time fake news identification in videos can be another possible future direction. Various detection techniques have been introduced by authors like 1. Linguistics basis Deception modelling, 2. Clustering, 3. Predictive modelling, 4. Content cue based methods and 5. Non text cue based methods.

[2] Although the Authors have compared various derivative methodologies, they have only shown accuracy of these models between 63 to 70 percent only.

[3] The authors have classified every tweet/post as binary classification Problem. The Classification is purely on the basis of source of the post/tweet. The Authors used manually collected data sets using twitter API , DMOZ . The following algorithms where used on data sets 1. Naïve Bayes, 2. Decision trees, 3. SVM, 4. Neural Networks, 5. Random Forest and 6. XG Boost. [3] The results show 15 percent fake tweets, 45 % real tweets , rest posts where undecided. These are models are text based only and have very little or negligible improvement on existing methods.

[4] The authors have introduced IEED for hoax detection based on Logistic Regression. They used Logistic Regression ML approach by combining news content and social content approaches. The authors Claim the performance is good as compared to described in literature. The authors implemented it with Facebook messenger chatbot. Three different datasets of Italian news posts on Facebook were used. Both content based methods with

social and content signals using Boolean crowdsourcing algorithms were implemented. [4] While the authors have extensively tested their approach on social media platforms, they have not stated the types of posts they have covered, their topics and their origins.

[5] The goal of the research presented in this paper is to identify false information by examining it in two stages: characterization and disclosure. In the initial phase, social media emphasises the fundamental ideas and tenets of fake news. In the discovery phase, various supervised learning algorithms are used to assess the existing techniques for fake news detection. XGboost, Random Forest, Naive Bayes, K-Nearest Neighbours (KNN), Decision Tree, and SVM are compared and SVM yields the highest accuracy of 92%. [5] The authors have focused their research on political articles and have neglected the possibility of other genres. Furthermore, they have not used algorithms like logistic regression which could yield better results compared to SVM.

[6] The paper proposes to use Genetic and Evolutionary Feature Selection (GEFeS) with the KNN machine learning algorithm to produce a 91% accuracy while detecting fake news. Additionally they use the GEFeS identified features and an optimal k value to train and test a quantum KNN (QKNN) to explore how quantum machine learning techniques can be utilized in fake news detection problems. The QKNN model archives 84% accuracy. Feature selection on average increased accuracy from 76 to 83 %. And from 83 to 87 on QKNN. [6] Theoretically quantum computers can do computation much faster than the computers in the server and our homes. They have also been proven to help solve problems which were impossible to solve using even supercomputers. In their model they got results for the quantum computer version of the algorithm which was lower. More algorithms should be developed which take leverage of the 4 states of quantum computers.

[7] This paper is very good to understanding the processes and options one has while tackling this field. They talk about data pre-processing with different methods to filter and clear data. In the second stage it tells us about the different methods like TF-IDF vectorizer, N-gram level vectorizer, Character level vectorizer etc to convert semantic data into its vector form. Finally it tells us about different classifiers like Random Forest, K-Nearest Neighbour, Linear Support Vector Machine etc. [7] Though the paper is good in informing us about the various methodologies it does not go deep into pros and cons of the algorithms which does not solve the problem to find the best possible solution to the problem.

[8] This paper also discusses more models to tackle fake news problem.They give us good ways to find datasets to train our model and mention CodaLab, ReCOVery, Fake and Real News etc. They talk about data cleaning and the structure of the data they applied punctuation remover stopword remover and porter stemmer to achieve this. The use three feature extraction algorithms Bag of Words, TF-IDF, and Word2Vec for feature extraction. Finally they allocated 67% data for testing and 33% for training and tried models like Multinomial Naive Bayes, MNB with Hyperparameter, Passive-Aggressive Classifier. [8] Although the authors do give us the stats until the data classification but do not give us much data on how each model performs.

[9] The authors use Decision Tree (DT) machine learning algorithm has to be worked out and also compare textual property accuracy with Support Vector Machine (SVM) machine learning algorithm. Finally a confusion matrix is made to get the stats of the data classification. [9] There is a gap in the existing systems that approached DT and SVM. Because it may be difficult to detect fake news with accuracy and precision. More values must be added to the dataset in order for the model to be trained to predict accurately.

[10] Neural networks have been proven to be one of the best ways to make ml and ai systems and this paper dives deeper into it. The author pre-processes the data extensively by removing fillers and using tokenization and lemmatization. Machine learning models can deal with numeric values only. Therefore, the author transforms the text input data into numeric vector form for applying classification methods like TF-IDF with N-gram Analysis and glove. Finally models are made using LSTM, FFNN, SVM, Neural Network etc. With the IOST benchmarks it shows SVM to work the best.

[11] Using several machine learning techniques, bogus news and posts may clearly be detected. Fake news is difficult to categorise because to its ever-changing traits and features on social media networks. However, computing hierarchical features is the primary distinguishing trait of deep learning. Numerous research projects are utilising deep learning techniques in a variety of applications, including audio and speech processing, natural language processing and modelling, information retrieval, objective recognition and computer vision, as well as multimodal and multi-task learning. These techniques include

convolutional neural networks, deep Boltzmann machines, deep neural networks, and deep autoencoder models.

[12] On Twitter, bogus news was primarily disseminated. Fake news mostly comes from social media. With the use of three different AI techniques and highlights extraction approaches, we have developed a discovery model for fake news in this work. The SVM classifier helps the suggested model achieve its most notable precision. The precision score of 95.05% is the highest.

[13] Finding out whether online news is accurate is important. The elements for identifying fake news are explored in the study. Be aware that not all phoney news will spread via online networking sites. SVM and NLP are currently being utilised to evaluate the suggested Naive Bayes classification algorithm. Future iterations of the algorithm could deliver superior outcomes using hybrid ways to achieve the same goals. Based on the models used, the system in question can identify bogus news. Additionally, it had offered some news suggestions on the subject, which is quite helpful to any user. In the future, the prototype's effectiveness and accuracy can be improved to a certain extent, and the user interface can also be improved.

[14] Different classification methods for public figure statements have been put into use. The suggested method makes use of classification algorithms including LR, RF, SVM NB, and DNN to help identify false news. For feature selection and extraction, classification approaches including LR, RF, SVM NB, and DNN are used. DNN works well in terms of execution speed and accuracy, but it requires more memory than other techniques. Then, we compare NB, RF, SVM, LR, and DNN based on time, memory, and accuracy. The comparison results show that DNN Algorithm is better than the rest algorithm in terms of accuracy and time kind because the rest classifiers require more time and give less accuracy thus DNN is more important to detect fake news.

[15] The best neural network model developed for this study can accurately and with few errors detect bogus news with up to 90.3% accuracy and 97.5% recall. Because N-gram vectors utilising TF-IDF will not only rely on term frequency but also a weight score that emphasises more significant terms, neural network models trained with N-gram vectors are performing somewhat better than models trained using sequence vectors. Due to their quick computation time and high recall rate, models trained using news titles are suited for usage in

social media apps where users are expected to respond quickly to any updates or incoming messages (low mistake rate). Any message that disseminates fake news can be halted with quick computation. To accurately identify fake news and prevent users from propagating it, models trained with news information that have higher accuracy and recall would be a better choice. On the other hand, for social media applications with feeds that are occasionally updated, rapid computation time is not very critical. The Keras neural network models can be further enhanced in the future by adjusting the parameters to attain even higher recall and accuracy. The performance of false news identification with NLP can also be improved by using a recurrent neural network (RNN) with a long short-term memory method (LSTM). Additionally, additional research can be conducted on news photos, videos, and articles to further enhance the models in the future. Besides that, to further implement this solution in Malaysia, similar approaches or techniques can be used to train the models with news dataset collected in Malaysia. Further research and experiment can also be done in Malay and Chinese news.

[16] The dataset was pre-processed by filtering redundant words. Feature extraction was applied and each term in the document was weighted for the construction of the Document Matrix. 23 different AI algorithms were applied on the data set to check the accuracy of each. By combining text mining methods and supervised artificial intelligence algorithms, a model was designed for the detection of fake news. The best mean values in terms of accuracy, precision, and F-measure were obtained from the Decision Tree algorithm.

[17] A model for detecting fraudulent news using several machine learning approaches was presented. The paper also looked at the four approaches and compared their accuracy. It is inferred that LSTM provides us with the highest accuracy (94%) followed by Keras Neural Network, SVM, and finally Naïve Bayes. The Keras based Neural network has a good accuracy that almost matches LSTM. LSTM provides such high results because the text is inherently a serialized object.

[18] This paper compares between the random forest and decision tree classification methods on a Kaggle based dataset It explains various pre-processing steps and concludes the solutions based on accuracy, recall and precision. The paper concludes that decision tree methods work better for all three performance measures, compared to the

random forest method. However, it also highlights the fact that proper processing also has an huge impact on the accuracy of both the methods.

[19] This paper proposes the use of two metaheuristic optimization algorithms namely GWO and SSO instead of classifiers in order to predict fake news. It also provides a detailed explanation on the working of both these optimization algorithms. As per its claim the paper proved that SSO and GWO would yield better results compared to other classifiers. Also GWO turned out to be the one with highest accuracy and precision in all the three datasets on which the training and testing was done.

[20] This paper explains about the different optimization techniques on the basis of their evolution, methodology, performance and applications. The techniques are swarm, bee colony, ant colony, genetic algorithm. All the algorithms are compared with each other. Genetic Algorithm is best because it may handle both continuous and discrete variable without any gradient information and it supports parallel computation.

[21] The paper aims to propose a feature selection based malware detection algorithm using Artificial Bee Colony (ABC). Done by selecting the most important and the most relevant features that are correlated with the seen and labelled data. The proposed algorithm enables researchers to decrease the feature dimension and as a result, boost the process of malware detection. The experimental results reveal that the proposed method outperforms the original.

[22] This paper propose a binary version of SSA and an improved version of SSA by using an inertia weight parameter to enhance algorithm. The maximum number of features is 856.the fitness function balances between the number of selected features and the classification accuracy. The proposed optimizer has outperformed the other optimizers in classification accuracy, being the second fastest optimizer and the one that has selected the minimal number of features. The results demonstrated that the integration of the inertia weight parameter in the SSA algorithm achieve better performance than other algorithms.

2.2 Summary/Gaps identified in the Survey

Fake news detection has been widely covered through various methodologies. A number of researches have been conducted in order to find the most efficient and accurate techniques for detecting fake news. However, each of them have their own significant shortcomings. In most of the research papers we've studied, the news data which contains the articles, which are used to train the models is insufficient and lacks quality and variability. Most methodologies utilise databases which only contain one single genre of news articles, most popularly, political news. This results into a less accurate model which is additionally only restricted to predicting the type of news that is fed into the machine learning model. Additionally, the researches lacks quality comparison reports amongst various binary classification models. Only a handful of researches have made in-depth comparisons and have unanimously concluded Logistic Regression and Support Vector Machines to be two of the most accurate binary classification machine learning models.

However, most approaches have used count vectorization as their method of representation of text data. Count vectorization creates a vector of word frequencies from a document. It generates a sparse matrix with each row denoting a document and each column denoting a distinct word in the corpus after counting the number of times each word appears in a given document. This method just takes word frequencies into account, disregarding word order, which is a major disadvantage when dealing with a large dataset. Another major disadvantage of count vectorization is that it treats all the words in the corpus equally and does not take into account their relative importance in an article. Hence after referencing to some approaches, we plan to use Term Frequency – Inverse Document Frequency (TF-IDF) method for representation of textual data. TF-IDF algorithm displays a document as a vector of word scores that illustrate the significance of words in the document in relation to the corpus as a whole. It considers the inverse document frequency (IDF) of a word over the entire corpus as well as the term frequency (TF) of that word in a given document. Words that are common in a document but uncommon in the general corpus are given greater weights by TF-IDF, suggesting their importance. This method lessens the effect of terms that are used frequently in various articles.

In conclusion, the papers that we've covered do not consider different genres of news. Political news has been utilised in most models as it is the easiest to interpret. Furthermore, we've determined that Logistic Regression and Support Vector Machine yields the highest mean accuracy and will be used in our model.

3. Overview of the Proposed System

3.1 Introduction and Related Concepts

i) TF-IDF:

The statistical technique known as Term Frequency - Inverse Document Frequency (TF-IDF) is frequently employed in information retrieval and natural language processing. It gauges a term's significance inside a document in relation to a corpus, or group, of documents. A text vectorization procedure converts words in a text document into significance numbers. There are numerous distinct scoring methods for text vectorization, with TF-IDF being one of the most used.

The TF-IDF vectorization/scoring method, as the name suggests, multiplies the Term Frequency (TF) and Inverse Document Frequency (IDF) of a word to determine its score.

Term Frequency: The Term frequency (TF) of a term or word is the ratio of the number of times the term appears to the total number of words in the document.

$$TF = \frac{Number\ of\ times\ the\ term\ appears\ in\ the\ document}{Total\ number\ of\ terms\ in\ the\ document}$$

Inverse Document Frequency: The IDF of a term indicates the percentage of corpus documents that contain the term. Words that are only found in a small number of papers, such as technical jargon terms, are given greater relevance ratings than words that are used in all publications, such as a, the, and.

$$IDF = \log\left(\frac{number\ of\ the\ documents\ in\ the\ corpus}{number\ of\ documents\ in\ the\ corpus\ contain\ the\ term}\right)$$

The TF-IDF of a term is calculated by multiplying TF and IDF scores.

$$TF - IDF = TF * IDF$$

ii) **Logistic Regression Model**: Logical regression is the best model for binary classification projects.

- Logistic regression falls within the category of supervised learning. Using a predetermined set of independent factors, it is used to predict the categorical dependent variable.
- Logistic regression forecasts a categorical dependent variable's output. As a result, the result must be a discrete or categorical value. It can be True or False, Yes or No, 0 or 1, etc., but rather than delivering an exact value between 0 and 1, it delivers probabilistic values that are in the range of 0 and 1.
- The main difference between linear regression and logistic regression is how they are used. While logistic regression is used to solve classification difficulties, linear regression is used to solve regression problems.
- In logistic regression, we fit a "S" shaped logistic function, which predicts two maximum values (0 or 1), rather than a regression line.
- The logistic function's curve shows the possibility of various events, such as whether or not the cells are malignant, whether or not a mouse is obese depending on its weight, etc.
- Because it can categorise fresh data using both continuous and discrete datasets, logistic regression is a key machine learning algorithm.
- Logistic regression may be used to categorise observations using a variety of data types, and it is simple to choose the best variables to employ.

iii) Logistic Function (Sigmoid Function):

- A mathematical function called the sigmoid function is employed to convert anticipated values into probabilities.
- It transforms any real value between 0 and 1 into another value.
- Since the logistic regression's value must lie within the range of 0 and 1, it can never go above or below this limit, resulting in a "S"-shaped curve. The sigmoid function or logistic function is another name for the S-form curve.

- We apply the threshold value idea in logistic regression, which establishes the likelihood of either 0 or 1. Examples include values that incline to 1 over the threshold value and to 0 below it.

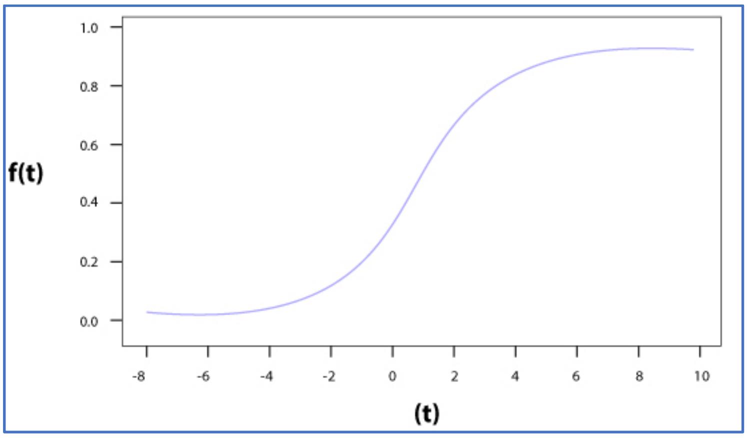


Figure 3.1: Sigmoid Function

3.2 Architecture for the Proposed System(with explanation)

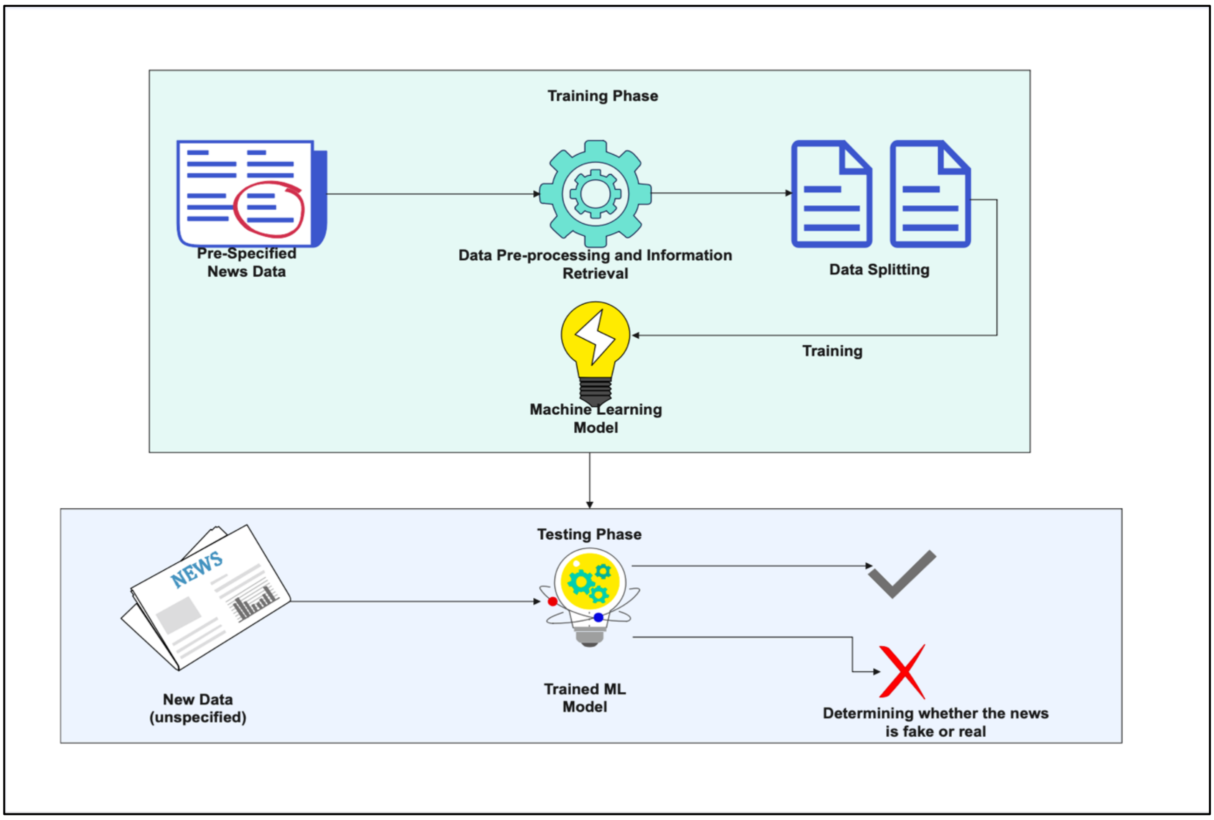


Figure 3.2: Architecture Diagram for Proposed System

i) **News Data:** news data is collected from various source with the attributes -

- id: unique id for a news article
- title: the title of a news article
- author: author of the news article
- text: the text of the article; could be incomplete
- label: a label that marks the article as potentially unreliable 1: unreliable and 0: reliable

This data is collected via various news publications and also social media sites.

ii) **Pre-processing Data:** Computers and systems don't understand text, they just understand characters plus the noise in the data should not be a lot the data should not face underfitting or overfitting. A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Data pre-processing is required for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.

iii) **Data splitting:** The data is split into 2 parts one for training and one for testing. We can also split the data into multiple parts to do k fold training to get better results.

3.2 Proposed System Model

Data Pre-Processing: Data pre-processing involves loading the dataset into pandas, evaluating the missing values from our dataset by using isnull().sum function, replacing missing values with appropriate values and separating the data and label (0 or 1) by storing them in different variables.

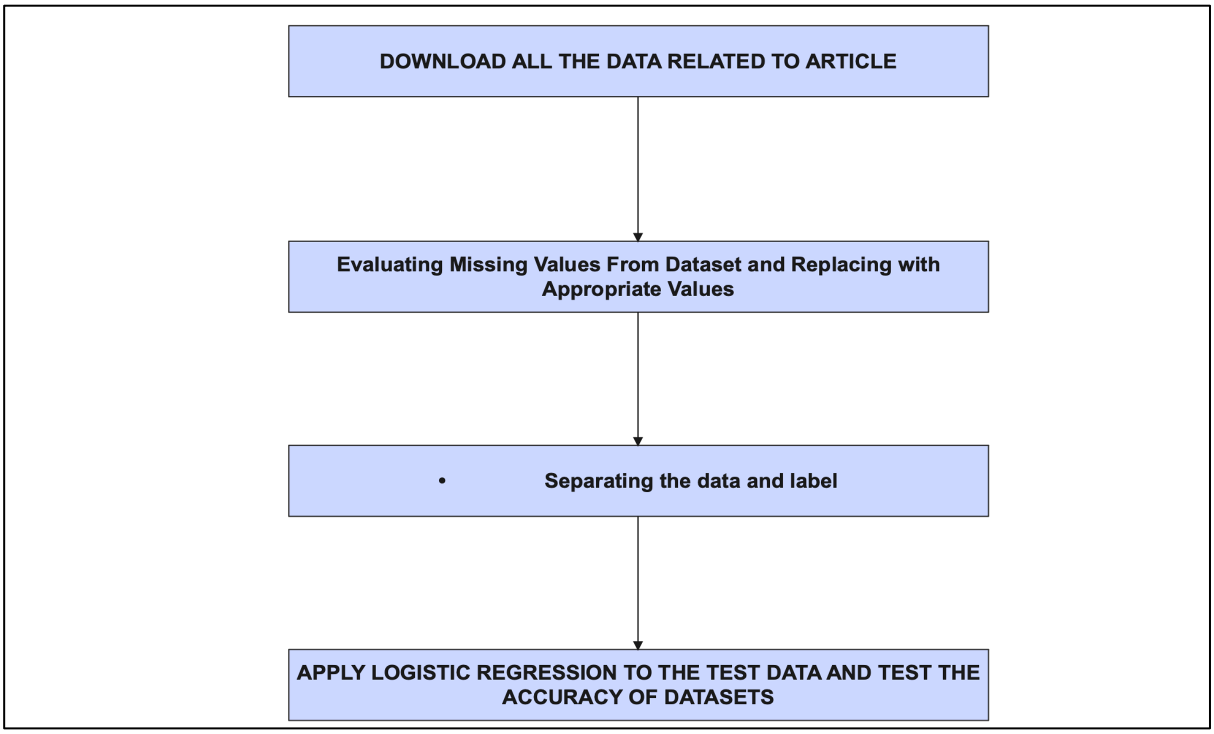

DOWNLOAD ALL THE DATA RELATED TO ARTICLE
Evaluating Missing Values From Dataset and Replacing with Appropriate Values
• Separating the data and label
APPLY LOGISTIC REGRESSION TO THE TEST DATA AND TEST THE ACCURACY OF DATASETS

Figure 3.3: Data Flow Diagram for Proposed System

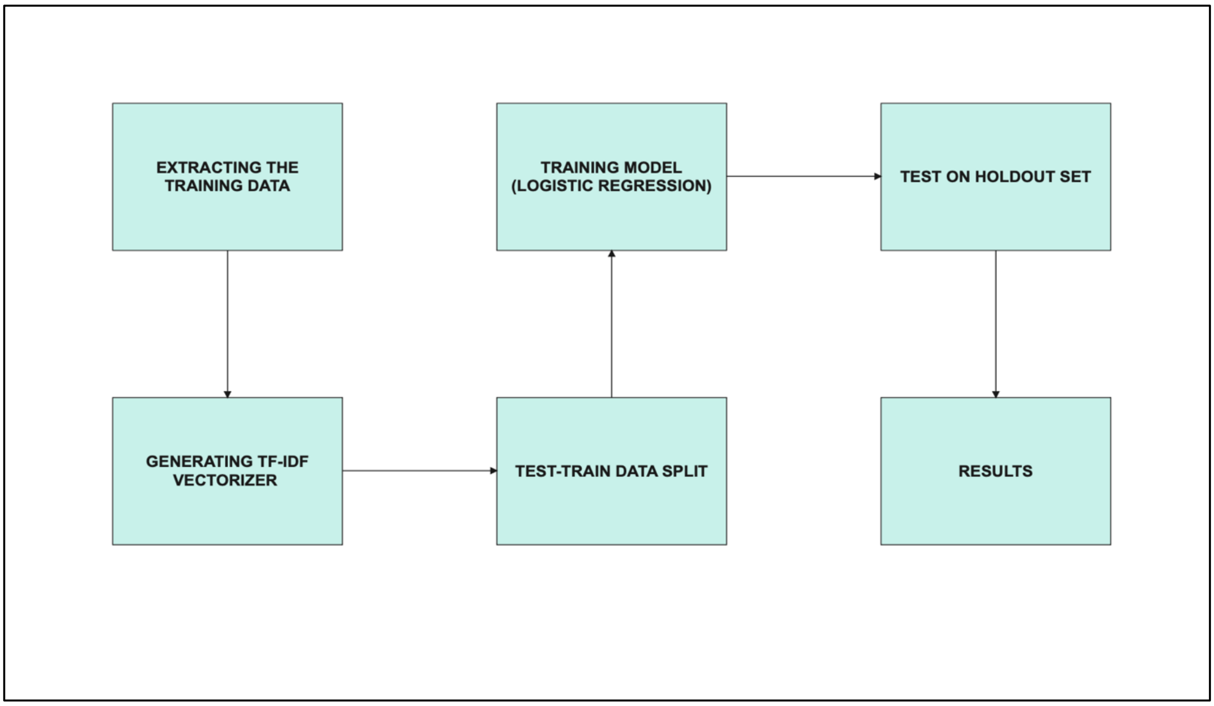

EXTRACTING THE TRAINING DATA
TRAINING MODEL (LOGISTIC REGRESSION)
TEST ON HOLDOUT SET
GENERATING TF-IDF VECTORIZER
TEST-TRAIN DATA SPLIT
RESULTS

Figure 3.4: Block Diagram for Data Pre-Processing

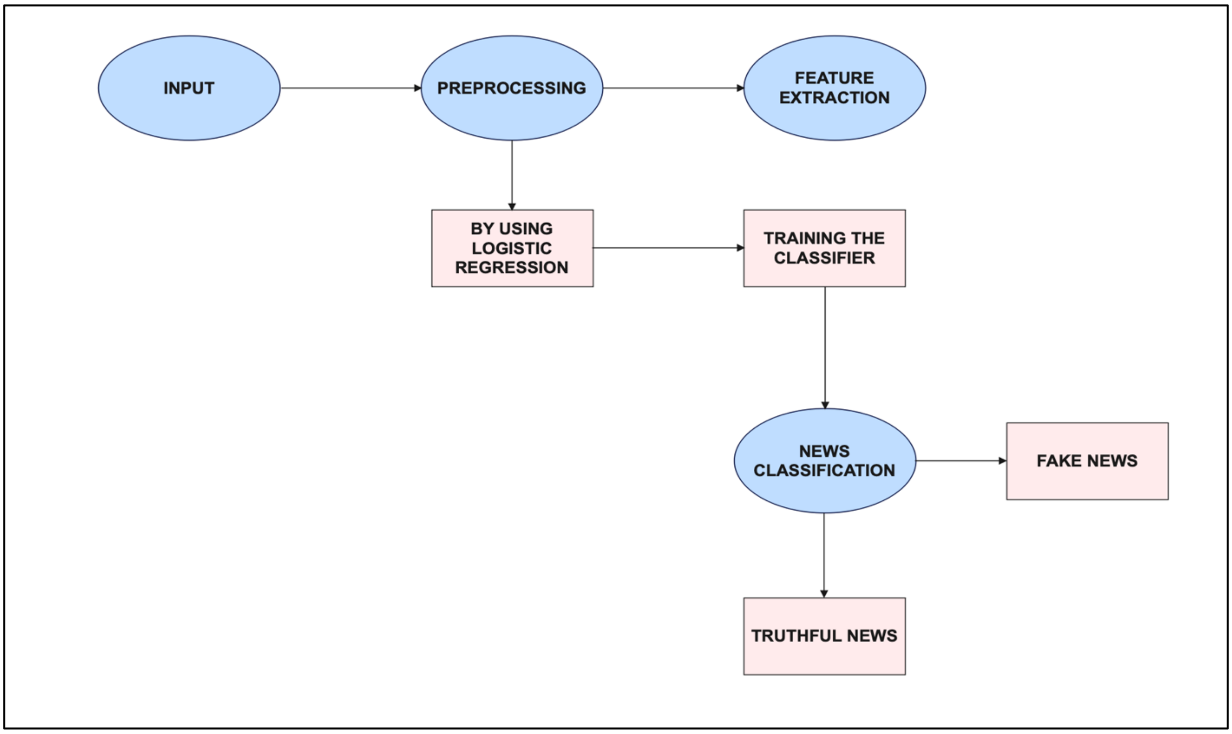


Figure 3.5: Data Flow Diagram for Data Pre-Processing

Activity Description for Proposed Model: Our proposed activity model for fake news detection application consists of a two-party system with a user and the system. The user register and then can log in with their credentials to the application. He/she can use the system for detecting whether a given article (input from the user) is fake or not.
The system on the other hand can also log in to the application but also has the authority over essential tasks like feature extraction and news data. The system can also change the classification of a given news article.

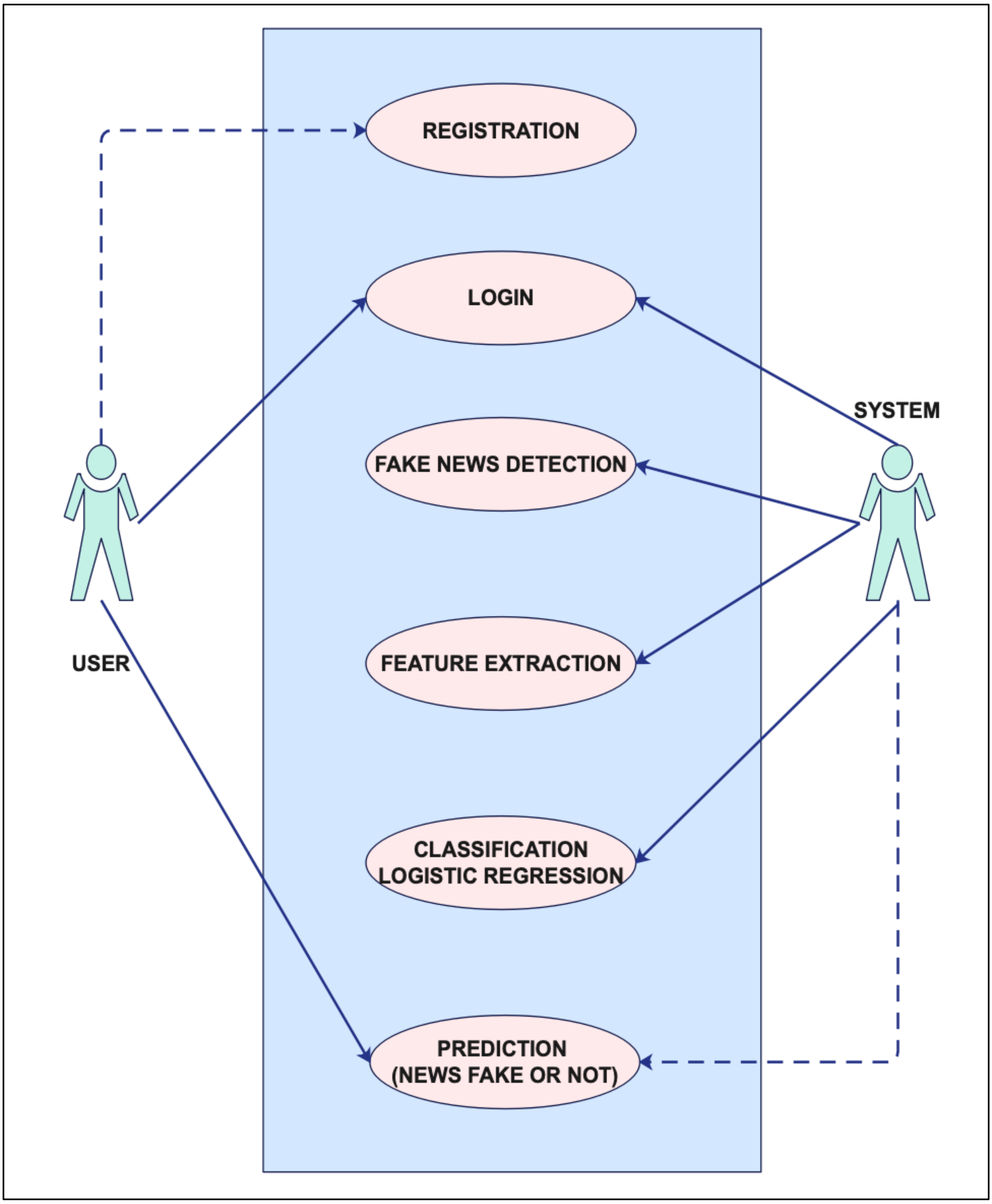


Figure 3.6: Activity Diagram for Proposed Model

4. Proposed System Analysis

4.1 Introduction

Through our project, we aim to develop a machine learning model that optimizes the detection of fake news by accurately classifying news articles as real or fake. This model should consider various features of the article, including language, sources, and sentiment, and learn from a large dataset of labeled examples. The goal is to achieve high accuracy and minimize false positives and false negatives, thereby reducing the spread of misinformation and ensuring that people have access to accurate and trustworthy information. The model should be scalable, adaptable to changing news landscapes, and easy to use for news organizations, social media platforms, and other entities.

We have divided the dataset into 80% training and 20% testing data. We have ensured accurate labelling of pre-specified news data as a false news detection system's performance depends on the precise labelling of training and test data. The importance of the sources used to gather the training and test data cannot be overstated. The accuracy and representativeness of the dataset are improved by the use of reliable and varied sources. A more thorough grasp of both actual and false news is attained by merging a variety of sources, such as reputable news organizations, fact-checking groups, and independent journalists. Using this strategy, the system will be trained on data that closely resembles real-world circumstances. Fact-checking and verification procedures are used to further support the reliability of the training and testing data. Further cross-referencing of the labelled articles and authenticity checks using external fact-checking agencies or APIs are possible. This recurrent method aids in locating any flaws or occurrences that were incorrectly labelled, enabling the dataset to be improved and refined. In addition to ensuring the correctness of the data, fact-checking encourages critical thinking and a culture of verification. The effectiveness of false news detection systems is directly impacted by the precision of the training and testing data. If the data contains examples that were incorrectly labelled, the system may develop biased or inaccurate patterns that impair its capacity to distinguish between true and false news. Inaccurate data can provide false positives or false negatives, which reduces the effectiveness of the system and erodes user confidence. As a result, acquiring high-quality, precisely labelled data is a crucial step in creating effective and trustworthy detection algorithms.

In conclusion, building efficient detection systems requires accurate training and testing data, which is essential in the fight against fake news. The performance and dependability of these systems are aided by accurate labelling, reliable sources, fact-checking, and consistent annotation. We can create effective false news detection systems that fight misinformation, shield people from manipulation, and safeguard the integrity of information in our communities by verifying the accuracy of the data.

4.2 Requirement Analysis

4.2.1 Functional Requirements

4.2.1.1 Product Perspective

One of the most challenging problems to overcome is analysing and identifying bogus news on the internet. Due to online media channels including social media feeds, blogs, and online newspapers, fake news has recently become a significant topic of discussion among the general public and researchers. 79 percent of people are concerned about what is real and false online, according to a BBC survey. Globescan performed the poll of more than 16,000 adults. "These poll results suggest that the age of 'fake news' may be as significant in lowering the credibility of online information as Edward Snowden's 2013 National Security Agency (NSA) surveillance revelations were in lowering people's comfort in expressing their opinions online," said Doug Miller, chairman of Globescan. After a bogus claim about Steve Jobs having a heart attack appeared on CNN's iReport, Apple's stock temporarily dropped by 10 points.

In India too, fake news can be abhorrently manufactured for political as well as personal gains. In a widely diverse nation like India, it can be used to spread propaganda against one single community, language, religion, etc. and can cause divide amongst people, and in some cases even riots and protests, which if turned violent can cause severe damage to both life and property.

For eg. social media was inundated with bogus news and distorted content as a result of the CAA Protests, which both targeted the demonstrators and Delhi police. Members of the governing BJP were observed circulating recordings that falsely suggested Aligarh Muslim University students were yelling anti-Hindu slurs. The Citizenship Amendment Act (CAA)'s benefits and goals were requested to be made public by the Supreme Court of India in order to weed out false information that was being spread about the topic. To indicate support for the act, BJP leaders distributed a phone number and asked people to leave a missed call. The phone number was extensively disseminated on Twitter along with fictitious promises of free Netflix subscriptions and company from lonely women.

Hence, a machine learning based automated tool for the detection of fake news will be useful for quick and reliable determination of news articles and will provide users with accurate results that will help them ignore the articles that are false.

4.2.1.2 Product Features

A fake news detection solution can be made more effective in recognising and flagging probable fake news pieces by including a number of features. Some of those features are:

- Article Analysis: Using natural language processing to examine the article's content and look for any suspicious trends, false facts, or biased language. To spot and rectify false news pieces, keep an eye out for contradictions, sensationalism, or inflated claims.

- Reputation Analysis of the Source: Analyzing the news source's standing and dependability. Aspects like the source's track record of accurate reporting, journalistic standards, fact-checking procedures, and any biases should be taken into account. Integrate third-party databases or APIs that offer insight into the reliability of news sources.

- Social Media Monitoring: Integration with social media platforms like Facebook, Instagram and Twitter to track the dissemination of articles and spot potential fake news using user feedback, engagement trends, or content-detection algorithms. Real-time fake news detection and reduction is made possible by this feature.

- User Feedback and Reporting: Permitting the users to flag content they believe to be fake news. Implement systems that allow users to comment on the veracity of articles that have been flagged so that the system can be improved over time. Taking user privacy into account and take action to stop misuse of the reporting tool.

- Fact-Checking Integration: Joining forces with reputable and reliable fact-checking agencies and news outlets to gain access to their databases or

APIs. Integrate the findings of their fact-checking efforts and draw on their knowledge to verify the veracity of publications or statements.

- Real-Time Updates: Updating the machine learning model's knowledge base on a regular basis to reflect new patterns in fake news, new sources, and cutting-edge disinformation strategies. Creating systems to keep the system current with the newest trends and methods for disseminating false information.

- API and Integration: Providing other platforms, like news aggregators, social media networks, news outlets, or browsers, the ability to use the fake news detecting features by providing an API or integration choices. This can limit the dissemination of false information on the internet at several touchpoints.

4.2.1.3 User Characteristics

While developing a fake news detection tool, it is crucial to take into consideration the characteristics and needs of the users who will interact with the system. This not only helps in catering to the needs of these individuals, but also helps to expand the customer base to wider audience while maintaining the accuracy and reliability of the tool. Some of the types of users that we have identified who can utilize the functionality of fake news are:

- General Users: These are regular people who read news and information from a variety of sources. They can lack in-depth understanding of methods for identifying false news or the capacity to assess the validity of news reports. The device must be simple to use, straightforward, and give unmistakable cues as to whether an item is likely to be a fake.

- Social Media Users: Users who primarily use social media platforms for news consumption. Fake news stories could be shared by their connections or appear in their news feed. As users peruse their news feeds, the solution should seamlessly integrate with social media platforms, providing real-time detection and alerting of possibly fraudulent news pieces.

- Journalists and News Professionals: These users contribute to the news' creation and distribution. According to client requirements, the product can offer extra features like access to reputable sources, tools for fact-checking, and source reputation research. It can help journalists preserve the integrity of their work by assisting with information verification prior to publication.

- Researchers and Fact-Checkers: Users who are actively analysing and verifying news stories in depth. Advanced features of the software may include access to extensive databases, integration with fact-checking organisations, and the opportunity to explore the in-depth research and supporting data behind an article's classification.

- Teachers, Professors and Students: Users working to advance media literacy and critical thinking abilities in school contexts. To help consumers comprehend the idea of false news, acquire detection procedures, and improve their capacity to assess the credibility of content, the product can offer educational materials, tutorials, and interactive activities.

- Policy Makers and Government Officials: Users who must consider how bogus news affects society and make wise decisions. The platform may offer thorough analyses, analytics, and insights on the prevalence and effects of fake news, assisting policymakers in creating effective countermeasures.

4.2.1.4 Assumptions and Dependencies

There are a few presumptions and dependencies that need to be taken into account while creating a false news detecting system. The system's capabilities and limits may be impacted by certain presumptions and dependencies. These presumptions and dependencies must be understood, and the system must be evaluated and updated on a regular basis to account for their effects. The effectiveness of the fake news detection system can be increased by identifying potential limits, addressing them, and taking user feedback into account.

These are as follows:

- Accessibility of Labelled Data: Systems for detecting fake news frequently use supervised learning techniques, which need a lot of tagged data to train the model. The system will need to be trained using a suitably big and representative dataset of labelled articles that can discriminate between authentic and false news. The system's efficiency may be impacted by the level of quality and diversity of the tagged data.

- Generalization of Training Data: The presumption is that the patterns and traits discovered from the training data will transfer well to other types of data, including changing misinformation strategies and false news pieces. However, if the system comes across novel or sophisticated bogus news that dramatically deviates from the training data, its performance may suffer.

- Accuracy of Labelling: The training data's labels for the articles are assumed to be accurate and trustworthy. Manual categorization, however, is prone to subjectivity, and various annotators may have different ideas about what counts as fake news. The performance of the system may be affected by incorrect labelling or inconsistent training data.

- Frequency of Updates: Systems for detecting fake news must be updated frequently to remain successful. Updates incorporating new sources, shifting strategies, and growing false news patterns are assumed to be easily accessible. The correctness of the system is dependent on frequent updates for dependencies on external sources, such as fact-checking databases or news source reputation APIs.

- Language and Cultural Context: Systems for detecting fake news could be reliant on certain linguistic and cultural environments. For example in our case, our system is solely designed to detect news articles that are written in the English language. Because of linguistic quirks, differences in writing styles, and cultural allusions, the system's efficacy can change between languages. Additional time and money are needed to adapt the system to various linguistic and cultural settings.

- Reliability of External Sources: Systems for detecting fake news may incorporate external sources for source reputation analysis or fact-checking. It is assumed that these outside sources are trustworthy and authoritative. Dependence on external sources, however, might have an effect on the system's overall performance because of the variable accuracy and dependability of these sources.

- Limitations of Text-based Analysis: Systems for detecting fake news generally examine the text of articles to look for patterns and characteristics that are specific to fake news. The idea is that language analysis by itself can offer trustworthy indicators of false news. Text-based analysis has its limitations, though, and it may be required to include contextual data, such as photographs, videos, or user engagement patterns, to increase the system's precision.

4.2.1.5 Domain Requirements

To ensure our model's usability, effectiveness and accuracy, we've considered numerous domain requirements which cater to specific needs of our fake news detection system. These are:

- Diverse as well as Reliable Data Collection: Assembling a trustworthy and varied dataset of articles (in our case, collected from kaggle.com and The Washington Post) with labels that represent both accurate and false news. To guarantee the system's ability to generalise across many settings, the dataset should encompass a wide range of topics, sources, writing styles, and formats.

- Pre-processing and Text Normalization: Pre-processing procedures should be used to clean and normalise the text data. To achieve this, it may be necessary to eliminate unimportant material (such as adverts and comments), deal with misspellings, inconsistent capitalization, punctuation, and punctuation, and normalise textual forms (such as converting URLs and numbers). We have removed all the null values from our dataset using the isnull().sum function.

- Feature Extraction and Representation: Obtaining useful features from text data that accurately capture the relevant traits of false news items. We have implemented TfidfVectorizer from sklearn.feature_extraction.text which is used to convert the text into feature vectors.

- Machine Learning Models and Algorithms: Choosing machine learning models and algorithms that are appropriate for detecting fake news. In our case, we have used Logistic Regression due to its high accuracy for binary classification applications, such as Fake News Detection. We have extensively studied various research papers to form a solid foundation for choosing logistic regression as our model and have compared it with other models like Support Vector Machines and K-Nearest-Neighbours.

- Training and Evaluation: Utilising the labelled dataset and the suitable training methods (such as cross-validation and stratified sampling), train the chosen models. Utilise pertinent assessment criteria to assess the models, such as area under the ROC curve, F1 score, recall, accuracy, and precision. Through iterative training and assessment cycles, continuously evaluate and improve the system's performance.

- Scalability and Performance: Ensuring that the system is capable of effectively handling a lot of articles. This entails streamlining the analysis and processing pipelines, taking into account distributed computing or parallel processing strategies, and efficiently allocating computational resources.

- Frequent Improvement and Maintenance: Establishing a procedure for the system's upkeep and ongoing improvement. Update the system frequently with new labelled data, consider and take into account user input, keep track of performance indicators, and deal with new problems and adjustments in the false news environment.

4.2.1.6 User Requirements

While developing our proposed system model for fake news detection, we had taken several steps towards catering all types of users who would be using our system to validate news articles. The fake news detection system can provide a user-centred experience that fulfils the demands of its intended users, improves their capacity to recognise fake news, and encourages responsible information consumption by attending to these user requirements. Some of those requirements we considered are:

- Easy Integration and Accessibility: We gathered that users favour a system that can be quickly incorporated into their current workflows or platforms, as well as being easily accessible. Accessible APIs or integration options should be made available by the system to enable easy integration into a wide range of applications, including web browsers, social media platforms, and news aggregators.

- Transparency and Explainability: Users want the system to be transparent and to be able to explain how it determines which objects to categorise. The traits or patterns that contribute to the classification of an item as potentially fraudulent should be explained by the system in a way that is both clear and understandable.

- Customization and Flexibility: We gathered that different types users may have different preferences or requirements for fake news detection. On one hand users can be general people who just want to validate whether a news article is fake or not, on the other hand users can be news organizations wanting to validate news before publishing it on their own platform to maintain reliability. The system should allow users to customize certain aspects, such as sensitivity thresholds, filtering criteria, or specific topics of interest. This customization empowers users to tailor the system to their specific needs.

- Educational Resources: Educational materials and advice on spotting false news, critical thinking, and media literacy may be useful to users, in particular educators and students. To help consumers improve their comprehension and awareness of bogus news, the system may provide lessons, advice, or links to outside educational resources.

- Performance and Scalability: Users anticipate the system to function well even in conditions of heavy traffic or article volume. The system must be scalable to accommodate rising user demand while maintaining peak performance.

- Real-Time Detection: Real-time or nearly real-time bogus news identification is valued by users. In particular, as news circulates quickly on social media sites, the system should be able to instantly analyse and categorise articles as they are encountered.

4.2.2 Non-Functional Requirement

4.2.2.1 Product Requirements

4.2.2.1.1 Efficiency (in terms of Time and Space)

The amount of features, the number of training examples, and the convergence criteria are just a few of the variables that affect how time- and space-complex logistic regression is. Here is a summary of the logistic regression's time and space complexity:

i) Time Complexity: The optimisation algorithm used to identify the ideal parameters often determines the time complexity of logistic regression during the training phase. Iterative techniques like gradient descent are the most widely utilised optimisation strategy for logistic regression. In logistic regression, the temporal complexity of gradient descent is frequently expressed as O(knd), where n is the number of training examples, d is the number of features, and k is the number of convergence iterations.
Logistic regression's temporal complexity during inference (i.e., predicting the result for brand-new examples) is typically minimal and can be regarded as constant or O(1). Only the dot product between the learnt parameters and the new instance's feature values and application of the logistic function to derive the anticipated probability are used in inference.
We simply need w and b to identify a line (in 2-D), plane (in 3-D), or hyperplane (in more than 3-D dimension) that can perfectly separate both the classes point so that when it encounters any new point, it can easily classify, from which class the unseen data point belongs. This is how we train a Logistic Regression model.
The values of w and b should be chosen so that the sum yi*wT*xi > 0 is maximised.
After training, we test our model with hypothetical data and determine its correctness. Understanding runtime complexity at that moment is

crucial. We obtain the parameters w and b following the training of the logistic regression.

We only need to execute the operation wT * xi in order to classify any new point. The point is positive if wT*xi > 0 and negative if wT*xi 0.

As shown earlier, performing the operation wT*xi requires O(d) steps since w is a vector of size d.

As a result, the Logistic Regression testing complexity is O(d).

As a result, Logistic Regression is excellent for low latency applications, particularly those where the data dimension is tiny.

ii) Space Complexity: The amount of features and the number of parameters that must be learned essentially determine the space complexity of logistic regression. The feature matrix for logistic regression has the dimensions n x d, where n is the quantity of training examples and d is the quantity of features. The feature matrix's storage requires O(nd) amount of space.

The parameter vector for logistic regression, which has dimensions d x 1, where d is the quantity of features, must also be stored. The parameter vector has an O(d) space complexity.

Given that the number of training instances n dominates the space needs in comparison to the number of features d, the overall space complexity of logistic regression is often O(nd + d) or O(nd).

During Runtime or Testing: After training the model what we just need to keep in memory is w. We just need to perform wT*xi to classify the points.

Hence, the space complexity during runtime is in the order of d, i.e, O(d).

4.2.2.1.2 Reliability

The methods and algorithms employed, the calibre of the training data, and the inherent difficulties involved in detecting fake news can all affect how reliable fake news detection systems are. It's critical to remember that detecting fake news is a complicated and developing field, and reaching complete reliability is difficult. Here are some factors to take into account when evaluating how reliable false news detecting systems are:

- Training Data Quality: For the system to be reliable, the training data's calibre is essential. Effective system training depends on high-quality, precisely labelled data that includes a variety of real and fraudulent news stories. However, the dependability of the system may suffer if the training data is inaccurate, biased, or contains labelling errors.

- Algorithmic Approaches: The accuracy of false news identification is impacted by the algorithmic methodologies and techniques chosen. Different strategies, including rule-based techniques, machine learning, and deep learning models, each offer advantages and disadvantages. The system's effectiveness and dependability relies on choosing the right algorithms and customising them to meet the demands of detecting fake news. We have used the logistic regression algorithm due to its high accuracy of predicting binary classification data.

- Dynamic Nature of Fake News: New methods and techniques are used to trick readers as fake news is continuously changing. Maintaining the detection system's dependability requires continuing to update and improve it in response to the changing landscape of fake news. Regular updates, adding new information, and modifying to new tendencies and tactics employed by disinformation marketers are critical actions.

- Feature Selection and Representation: For false news identification to be accurate, informative elements must be carefully chosen and presented. It is crucial to select pertinent variables that capture the distinctive qualities of fake news, such as linguistic patterns, content sources, and contextual indicators. To ensure appropriate categorization, the representation of these characteristics needs to be strong and take into account the subtleties of fake news.

- Generalization: A false news detection system must be able to transfer the knowledge it learns from training data to other situations. A trustworthy system should perform effectively on both new and previously unread articles in addition to the training data. The system's generalisation capabilities may be determined by analysing how well it performs on various datasets and in various domains. We have also ensured our system can predict all genres of news articles, something most systems we have studied lacked.

- User Feedback: The development and assessment of fake news detection systems can be improved by incorporating human experts, fact-checkers, and topic experts. Expertise and judgement from humans can enhance the capabilities of automated systems and offer insightful information. Additionally, soliciting user feedback, keeping track of system performance, and incorporating user viewpoints can assist find possible problems and improve system dependability.

In addition, we have also considered the following reliability aspects while developing our system:

- The model should not have a political bias. If the model suffers from any political bias and is not based on facts it will not be trusted to be used.

- The model should be accurate and must have a correction mechanism. This mechanism will be a feedback system. The users can file a simple complaint with us on the article where they find out the model was not accurate and then humans will verify the facts and feed in the data to update the model.

- The model must be continuously updating.

- The system must be secure and must not have loopholes which could be used to tamper with the model.

- The system must have a very low mean time between failures (MTBF).

- The system must be able to recover from any failure within 5 minutes.

- The system must have an error logging and reporting mechanism to facilitate debugging and maintenance.

- The system must comply with govt norms.

- The system must comply with the relevant safety and security standards.

- Parts of the system should be decoupled and the code should be well written and maintained.

4.2.2.1.3 Portability

When discussing fake news detection, portability refers to a detection system's capacity to be applied and used in many situations, platforms, or settings. Portability is a crucial factor since it enables the false news detection system to be used in a variety of scenarios. Aspects of false news detection portability include the following:

- Scalability: Scalability and portability go hand in one. The workload that a portable false news detection system can handle varies, and it should be able to scale up or down according to demand. This scalability enables the system to process enormous numbers of news articles or user-generated material efficiently and adjust to changing requirements.

- Minimal Resource Requirements: The minimum compute, memory, and storage requirements for a portable system should apply. As a result, the system may be effectively installed and operated on a variety of hardware and infrastructure, from low-power devices to high-performance servers.

- Cross-platform Compatibility: Various platforms, including online browsers, mobile apps, social media platforms, and content management systems, should be supported by a portable fake news detecting system. Because of its interoperability, the system can be easily integrated into many systems without requiring major alterations or adaptations.

- Flexible Deployment Options: Flexible deployment choices, such as on-premises, in the cloud, or as a Software-as-a-Service (SaaS) solution, are made possible by portable systems. Users can select the deployment strategy that best meets their goals, infrastructure, and resources thanks to this flexibility.

- Adaptability to Different Domains and Languages: A portable system need to be able to accommodate various linguistic and domain contexts. A portable system should be able to properly handle a variety of topics, sources, and languages because fake news detection is not restricted to a single area or language.

The model is a standalone entity and could be integrated with any tool out there. The model can be deployed on the server or can also be run on a local machine. It is not a very big model and as nowadays 5-6 GB of data is not a big issue the model is very portable.

Requirement to run the model are -

- The web application must be compatible with the latest versions of Chrome, Firefox, Safari and Edge browsers.
- The software system must be dockized and containerized so it is deployable on AWS, Azure, Google Cloud platforms or any other cloud platform or server.

Requirements to run the model locally are -

- The desktop application must run on Windows 10, MacOS and Linux operating systems.
- The mobile application must support Android and iOS devices.
- Having at least 10 GB of free storage
- Having a decent modern processor.

Requirements to run the model as a developer are -

- Having google collab or any other python notebook app.
- Having skills in python.
- Should have git for version control.

4.2.2.1.4 Usability

- The user interface must follow the principles of consistency, simplicity, feedback, and visibility.

- The user interface should have a basic input for inputting news and a clear output to show if it seems to be real or fake.

- The system should have a form to file inaccuracies.

- The user interface must provide clear and concise instructions, labels, and error messages.

- The user interface must support accessibility features for the disabled.

- A fake news detecting system must have an intuitive UI. Users should be able to explore and interact with the system with ease thanks to its user-friendly and visually appealing interface. A positive user experience is facilitated by clear and succinct instructions, well-designed menus, and aesthetically instructive displays.

- A fake news detection system must respond quickly to user expectations. The system should offer real-time analysis and promptly transmit results. The user experience can be hampered by slow or delayed responses, which may deter consumers from using the system.

- The system should clearly and simply display the findings of the fake news study. An indication of the article's credibility, noting whether it is true, fraudulent, or uncertain, should be included in the output. The results' capacity to be understood can be improved by providing more details or explanations about the elements that went into the classification.

- Accessibility issues should be taken into account in the system to ensure that people with impairments can use it efficiently. This covers keyboard accessibility, giving alternative text for images, and being compatible with assistive technologies.

4.2.2.2 Organizational Requirements

4.2.2.2.1 Implementation Requirements (in terms of deployment)

A system for detecting fake news must carefully take into account a number of essential parameters. Here are some crucial implementation criteria to consider:

- Collecting examples of both true and false news in a varied and representative sample of labelled news stories. To successfully train the detection model, make sure the dataset is diverse in terms of themes, sources, and situations. Data collection techniques include using existing datasets, fact-checking partnerships, and scraping news websites.

- Feature engineering is done to improve the detection system's performance. Incorporating domain-specific information, choosing useful features, dimensionality reduction, or feature modifications may all be necessary to achieve this. Techniques for feature engineering assist in enhancing the system's capacity to capture pertinent signals and reduce noise.

- Dividing the dataset into training and validation sets to test the detection model's effectiveness. Use cross-validation methods to evaluate the model's generalizability and avoid overfitting, such as k-fold cross-validation. Train and improve the model iteratively to maximise performance and guarantee accurate detection results. We have split our dataset into 20% testing and 80% training data.

Method 1

The software is to be deployed serverless functions on the cloud which fire up a new process each time a request has been made. This allows the application to be infinitely scalable and highly available. Serverless also prevents the use of load balances and gives high concurrency and parallelism.

Proper VPC should be set up. VPC or virtual private cloud will separate the model from the internet and will allow access only to the IPs which are whitelisted in the NAT gateway.

Method 2

The model can be made as a plugin to chatGPT which makes it run on top of chatGPT as an extension. This does not require any hosting from our side. The user can just download the model and add it to his extensions. The model will run on the user's computer and regular updates will be sent.

4.2.2.2.2 Engineering Standard Requirements

- The model should follow python PIP 8 linting so the team has consistent code.
- Formatting of code should be defined at the beginning and followed through with the use of auto formatter so the team has consistent code.
- The code should be componentized and decoupled so that the components are reusable and replaceable.
- The programming should be done to interfaces and not the implementation.
- The data used for training or testing the ML model must be accurate, complete, consistent, relevant, etc
- Proper design patterns should be followed and documented.
- Proper system designs should be created and documented.
- The data used for training or testing the ML model must be protected from unauthorized access, disclosure, or misuse
- The degree to which the ML model produces correct predictions or decisions for a given input must be high.
- The ML model should handle variations or uncertainties in the input data or the environment.
- The system should be performant and produce fast results.

4.2.2.3 Operational Requirements (Explain the applicability for your work w.r.to the following operational requirement(s))

i) Economic

The system should operate within an efficient multi talented team which is highly motivated to make people free from false information. Various costs associated with the project are -

- Development cost : Cost for continuous development of the model, Cost for development for the frontend and backend, extension development will require additional developer, and also dev-ops. Apart from the actual development of the product some companion teams for security, performance, and scalability needs to be added. Managers to manage all these human resource will require additional costs.

- Operational cost : running up the website, distribution on the extension, manual verifiers for the complaints about the results, dev team to better the model. A dedicated operations team for govt compliance is needed. Advertising will play a major role in making people aware.

- Business model : The website will be free to use for individual verifiers and will generate revenues through ads. The main earnings will be a B2B model where we will sell the model's api access to various businesses.

ii) Environmental

The system should operate in different physical environments and conditions[1]. For example, the system should be able to withstand extreme temperatures, humidity, dust, vibration, etc.

- Environmental compatibility : The software is well suited for any environment where a computer can work and an internet connection is available.

- Environmental impact :The software has no direct environmental impact but indirectly, the software requires processing power which definitely uses energy thus it does have an environmental impact which should be thought about. The hardware running the software is also made using mined materials.

- Environmental compliance : No environmental compliance is required.

iii) Social

The system should perform in a socially acceptable and respectful manner. It should not be very confrontational, rather it should take a more educational approach.

- Social compatibility : The software is well suited for any social community where a computer can work and an internet connection is available. People can easily check for the correct information. The system shall comply with the Web Content Accessibility Guidelines (WCAG). The system should also provide a user friends experience based on usability testing.

- Social impact : The software makes people be more aware about the information they are consuming and helps them differentiate between truth and false much better.

- Social compliance : Indian IT act is applicable.

iv) Political

The software should not try to interfere directly into politics. Though it will influence politics and fake info is used to sway voters by all parties.

- Political compatibility : The software is well suited for any political community. People can easily check for the correct information about politics and make better judgements.

- Political impact : The software makes people be more aware about the information they are consuming and helps them differentiate between truth and false much better. Thus they are much aware about the political landscape and make much better decisions.

- Political compliance : Data and copyright laws do apply in addition to the IT act.

v) Ethical

The system should not think about ethics and try to make moral judgements. This is a fact based product and must stick to correcting facts.

- Ethical compatibility : There is an ethical dilemma about what is fake news because many times true and false are just perspectives. The model should focus on facts as much as possible.
- Ethical impact : The software makes people be more aware about the information they are consuming and helps them differentiate between truth and false much better. Thus they are much more aware about the environment and thus make better decisions.
- Ethical compliance : Indian IT ACT is applied.

vi) Health and Safety

The software should not cause any health and safety concern.

- Health and Safety compatibility : There is no health and safety concern about the software.
- Health and Safety impact : The software does not impact the health and safety of any user.
-
- Health and Safety compliance : Indian IT ACT is applied.

vii) Sustainability

The system should be sustainable over the long term. It should be kept updated regularly. The devops should be scalable and should not cost the business a lot of resources.

- Sustainability compatibility : The software can function in any case where a modern computer is present with active internet connection.
- Sustainability impact : The software makes society much more aware of facts and thus makes the society much more sustainable.
- Sustainability compliance : Indian IT ACT is applied.

viii) Legality

The system should be legally compliant and follow the law of the land. It should be properly reviewed to fend of any breaches.

- Legality compatibility : The software follows Indian IT ACT and is fully compatible with the law of the land.
- Legality impact : New laws on data protection and copyright might hamper the product as then the data will not be for anybody to use.
- Sustainability compliance : Indian IT ACT is applied.

ix) Inspectability

The system should have proper regular audits with guided inspections to test sustainability, scalability, fault tolerance, political bias, accuracy etc.

- Inspectability compatibility : The software is open course and can be inspected at any time by any organization and even common folks.
- Inspectability impact : The software does not impact inspectability.
- Sustainability compliance : Indian IT ACT is applied.

4.2.3 System Requirements

4.2.3.1 H/W Requirements(details about Application Specific Hardware)

- System - Pentium-IV
- Speed - 2.4GHZ
- Hard disk - 40GB
- Monitor - 15VGA colour
- RAM - 512MB

4.2.3.2 S/W Requirements(details about Application Specific Software)

- Programming language used: Python
- Platform used: Google Colab
- Libraries used: Numpy, pandas, re, stopwords from NLTK(Natural Language Toolkit), PorterStemmer from nltk.stem.porter, train_test_split from sklearn.model_selection, LogisticRegression from sklearn.linear_model and accuracy_score from sklearn.metrics.
- Dataset Used: https://www.kaggle.com/c/fake-news/data?select=train.csv
- Software Used for Figures: Wondershare EdrawMax

Software Environment – PYTHON

- Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.
- Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- Python is Interactive − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- Python is a Beginner's Language − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

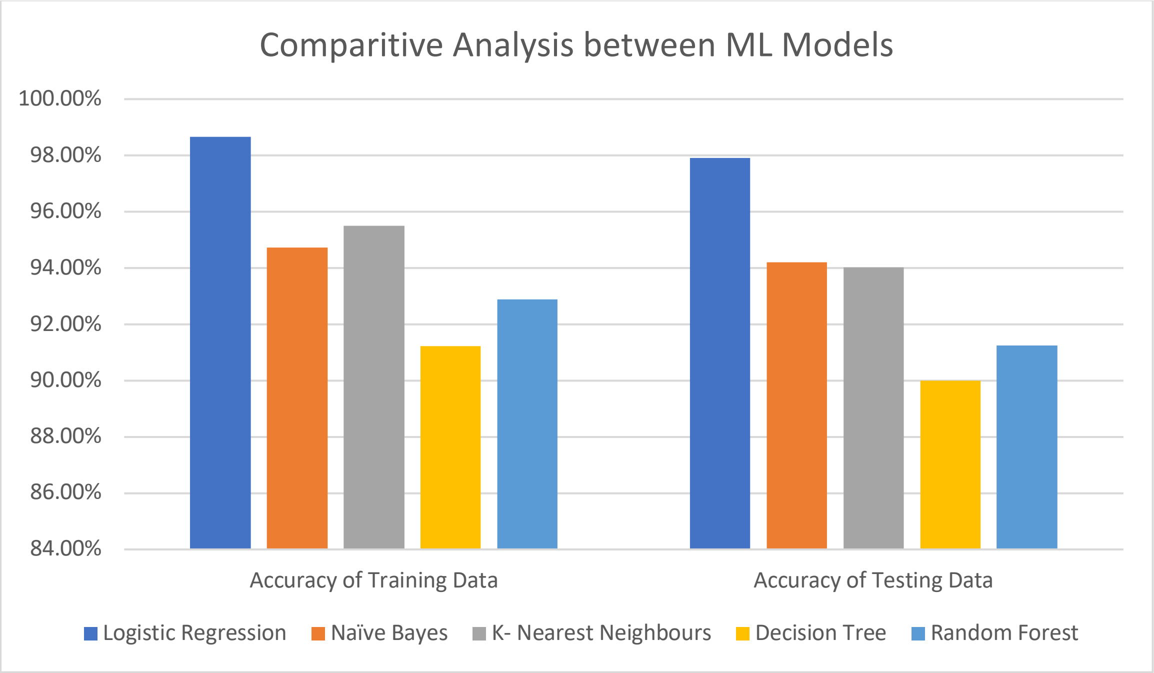
5. Results and Discussion

I) Comparative Analysis:

We started by comparing the following commonly used machine learning models for binary classification on our dataset using Google Collaborate to determine which algorithm is the most accurate for our system.

Table 5.1: Comparative Analysis of ML Models

Accuracies	Logistic Regression	Naïve Bayes	K- Nearest Neighbours	Decision Tree	Random Forest
Accuracy of Training Data	98.65%	94.73%	95.49%	91.22%	92.88%
Accuracy of Testing Data	97.90%	94.20%	94.02%	90%	91.24%



As inferred from the results above, we can clearly observe Logistic Regression is the most accurate ML model both in terms of training and testing accuracies. Hence, we will proceed with Logistic Regression for our Fake News Detection system.

Methodology:

a) We start with creating new project file in Google Collab which can be predominantly used to execute Python and Terminal commands.

b) Next, we import the libraries listed as below:

- numpy: useful for making numpy arrays
- pandas: useful for creating the data frames and storing the data in the data frame
- re: Regular expression, useful for searching words in a text or paragraph
- stopwords from NLTK(Natural Language Toolkit): stopwords are words that do not add value about the context of the data to the paragraph (eg: a, an, the, where, what, why etc)
- PorterStemmer from nltk.stem.porter: Used to stem our words
- TfidfVectorizer from sklearn.feature_extraction.text: Used to convert the text into feature vectors.
- train_test_split from sklearn.model_selection: Used to split our dataset into training and test data.
- LogisticRegression from sklearn.linear_model
- accuracy_score from sklearn.metrics

c) Downloading the English stop words from NLTK library.

d) Data Pre-Processing:

- Loading the dataset to pandas data frame

- Evaluating the missing values from our dataset by using isnull().sum. This calculates the missing values in each column.

- Data processing to replace missing values with appropriate values.

- Replacing the null values with empty string.

- For increasing the accuracy of our prediction model, we use the titles of the articles and the authors of the articles since using the texts which are large in size may yield a higher rate of false predictions. This also reduces the processing time of our prediction model.

- Separating the data and label(0 or 1) by storing them in different variables using the drop() function.

e) Stemming:

Stemming is the process of reducing a word and extract its root word or its base. This is done by removing the prefixes and suffixes of the words. Eg.: Act→ acting, actor, actress. This will be helpful in reducing the total amount of words from our dataset in order to decrease the processing time. This will be performed by using the PorterStemmer function from nltk.stem.porter.

- Creating a function called stemming with input as authors and titles of the articles.

- Using re.sub to gather all the alphabets (both lower and upper case) from the dataset by removing all the numbers, punctuations and special characters.

- Converting all the words of the dataset to lower case by using the .lower() function as our machine learning model may think the upper case words may mean more significant than lower case ones.

- Splitting and converting words to lists by using the .split() function.

- Applying .stem() function to each of the words in the dataset except the stop words.

- Returning the processed text.

- Applying this stemming function to the dataset.

- Separating the data and label by storing them in two separate integers X(data) and Y(label).

- Converting the textual data to numerical data as machine learning model can not understand textual data. This is done by using TFIDF vectorization.

- TfidfVectorizer() function counts the number of times a word is resent in the textual data.

f) Splitting:
In the next step we split our dataset into training and test data.

- Four variables are created (X_train, X_test, Y_train and Y_test).

- .train_test_split function is used. Test size is taken as 0.2. Hence test data is 20% and training data is 80%.

g) Training:

Training our machine learning model using Logistic Regression.

- .fit() function is used with the parameters X_train and Y_train to train our model.

- .fit() function plots the sigmoid curve using logistic regression.

h) Evaluation

- Accuracy scores are determined for training data by using model.predict() function with X_train as parameter.
- Accuracy scores are determined for test data by using model.predict() function with X_test as parameter.

h) Prediction:

The final step involved is making a predictive system if the accuracy scores are satisfactory.

- A value from X_test data is randomly chosen and is fed into a new variable X_new.

- This test data is excluding the label indicating whether the news is true or false. Hence the data being put in the predictive model is unspecified.

- Model.prediction() function is used to derive the label of the data through the logistic regression predictive model.

- If the label derived is 1, the news is printed as fake news, otherwise it is true.

- Lastly, the predicted value through the model is compared with Y_test of the same data which contains the original label values.

Result:

After applying Logistic Regression combined with the information retrieval technique TF-IDF vectorizer, we have achieved the following accuracies from our model:

Training accuracy: 98.65%
Testing accuracy: 97.9%

6. References

Weblinks:

https://www.kaggle.com/c/fake-news/data?select=train.csv (Accessed on February 2023)

Journals:

1. Ahmad, I., Yousaf, M., Yousaf, S., & Ahmad, M. O. (2020). Fake news detection using machine learning ensemble methods. Complexity, 2020, 1-11.

2. Parikh, S. B., & Atrey, P. K. (2018, April). Media-rich fake news detection: A survey. In 2018 IEEE conference on multimedia information processing and retrieval (MIPR) (pp. 436-441). IEEE.

3. Helmstetter, S., & Paulheim, H. (2018, August). Weakly supervised learning for fake news detection on Twitter. In 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM) (pp. 274-277). IEEE.

4. Della Vedova, M. L., Tacchini, E., Moret, S., Ballarin, G., DiPierro, M., & De Alfaro, L. (2018, May). Automatic online fake news detection combining content and social signals. In 2018 22nd conference of open innovations association (FRUCT) (pp. 272-279). IEEE.

5. Khanam, Z., Alwasel, B. N., Sirafi, H., & Rashid, M. (2021, March). Fake news detection using machine learning approaches. In IOP conference series: materials science and engineering (Vol. 1099, No. 1, p. 012040). IOP Publishing.

6. Manzoor, S. I., & Singla, J. (2019, April). Fake news detection using machine learning approaches: A systematic review. In 2019 3rd international conference on trends in electronics and informatics (ICOEI) (pp. 230-234). IEEE.

7. Abdulrahman, A., & Baykara, M. (2020, December). Fake news detection using machine learning and deep learning algorithms. In 2020 International Conference on Advanced Science and Engineering (ICOASE) (pp. 18-23). IEEE.

8. Nath, K., Soni, P., Ahuja, A., & Katarya, R. (2021, August). Study of fake news detection using machine learning and deep learning classification methods. In 2021 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT) (pp. 434-438). IEEE.

9. Krishna, N. L. S. R., & Adimoolam, M. (2022, February). Fake News Detection system using Decision Tree algorithm and compare textual property with Support Vector Machine algorithm. In 2022 International Conference on Business Analytics for Technology and Security (ICBATS) (pp. 1-6). IEEE.

10. Hlaing, M. M. M., & Kham, N. S. M. (2021). Comparative Study of Fake News Detection Using Machine Learning and Neural Network Approaches. In International Workshop on Computer Science and Engineering (pp. 59-64).

11. Manzoor, S. I., & Singla, J. (2019, April). Fake news detection using machine learning approaches: A systematic review. In 2019 3rd international conference on trends in electronics and informatics (ICOEI) (pp. 230-234). IEEE.

12. Ahmad, I., Yousaf, M., Yousaf, S., & Ahmad, M. O. (2020). Fake news detection using machine learning ensemble methods. Complexity, 2020, 1-11.

13. Jain, A., Shakya, A., Khatter, H., & Gupta, A. K. (2019, September). A smart system for fake news detection using machine learning. In 2019 International conference on issues and challenges in intelligent computing techniques (ICICT) (Vol. 1, pp. 1-4). IEEE.

14. Hiramath, C. K., & Deshpande, G. C. (2019, July). Fake news detection using deep learning techniques. In 2019 1st International Conference on Advances in Information Technology (ICAIT) (pp. 411-415). IEEE.

15. Kong, S. H., Tan, L. M., Gan, K. H., & Samsudin, N. H. (2020, April). Fake news detection using deep learning. In 2020 IEEE 10th symposium on computer applications & industrial electronics (ISCAIE) (pp. 102-107). IEEE.

16. Ozbay, F. A., & Alatas, B. (2020). Fake news detection within online social media using supervised artificial intelligence algorithms. Physica A: statistical mechanics and its applications, 540, 123174.

17. Reddy, P., Roy, D., Manoj, P., Keerthana, M., & Tijare, P. (2019). A Study on Fake News Detection Using Naïve Bayes, SVM. Neural Netw. LSTM J Adv Res Dyn Control Syst, 1, 942-947.

18. Jehad, R., & Yousif, S. A. (2020). Fake news classification using random forest and decision tree (j48). Al-Nahrain Journal of Science, 23(4), 49-55.

19. Ozbay, F. A., & Alatas, B. (2019). A novel approach for detection of fake news on social media using metaheuristic optimization algorithms. Elektronika ir Elektrotechnika, 25(4), 62-67.

20. Tyagi, K., & Tyagi, K. (2015). A comparative analysis of optimization techniques. Int. J. Comput. Appl, 131(10), 6-12.

21. Mohammadi, F. G., Shenavarmasouleh, F., Amini, M. H., & Arabnia, H. R. (2020, September). Malware detection using artificial bee colony algorithm. In Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers (pp. 568-572).

22. Hegazy, A. E., Makhlouf, M. A., & El-Tawel, G. S. (2020). Improved salp swarm algorithm for feature selection. Journal of King Saud University-Computer and Information Sciences, 32(3), 335-344.