Fake news detection using machine learning

Ojas Bhat ,Mrudula Pulidindi ,Suha ,Satesh Kumar

School of Computer Science and Engineering, VIT-AP University, Guntur, India

Emails: ramkrishna.22bce8098@vitapstudent.ac.in , mrudula.22bce8123@vitapstudent.ac.in , suha.22bce79284@vitapstudent.ac.in, Satesh.22bce7914@vitapstudent.ac.in ,

1.Abstract

Fake news detection has emerged as a critical challenge in the digital age, where misinformation can spread rapidly through social media and online platforms. This project aims to develop an automated system that employs machine learning techniques to classify news articles as either genuine or fabricated.

The problem of fake news detection has become increasingly critical due to the rapid dissemination of false information through social media and online platforms.[3] This challenge is characterized by several unique aspects that complicate the detection process.

While social media provides low-cost and rapid access to information, it also facilitates the widespread sharing of low-quality news, which can have detrimental effects on public opinion and societal trust.[3] Future research should focus on integrating diverse data sources and improving the robustness of detection algorithms to effectively combat the spread of misinformation in digital media.[4]

The system's primary features include real-time content classification, source verification, and a user feedback mechanism, enabling continuous improvement. Core machine learning algorithms—such as Support Vector Machines and Neural Networks—will be employed for high accuracy, while a scalable architecture will ensure the system can handle large volumes of data from high-traffic sources.[1] Designed for ease of use, the system provides immediate feedback on news authenticity, helping to mitigate the harmful impact of misinformation and promote informed decision-making.

2.INTRODUCTION

Fake news often contains misleading information that can be fact-checked, such as inaccurate statistics or exaggerated service costs, which can lead to unrest, as seen during many times in history.[1] While organizations work to address these issues by holding authors accountable, their efforts are limited[5]. They rely on manual detection, which is not scalable given the vast number of articles published globally every minute.

A potential solution is to develop a system that provides an automated credibility index or rating for publishers and news content.[1] This paper proposes a methodology to create a model that detects whether an article is authentic or fake based on its words, phrases, sources, and titles.[2] By applying supervised machine learning algorithms to a manually labelled dataset, feature selection methods will be used to identify the most relevant features for achieving the highest precision, as measured by a confusion matrix.[2] The model will be trained using various classification algorithms, and once tested on new data, the final product will be a tool that can detect and classify fake articles, capable of being integrated into other systems for future use.

This project provides a detailed specification of the Fake News Detection System, covering aspects such as functional requirements, system architecture, data requirements, and performance constraints. Key features, like

NLP-based text processing and the integration of popular machine learning algorithms (e.g., Support Vector Machines, Decision Trees, and Neural Networks), will be utilized to ensure accurate classification.[3]

By providing real-time detection and classification, this system will enable users to:

1. Identify Potentially Misleading Content: Users will have access to a credibility score for each news item, helping them discern legitimate information from misinformation.[3]
2. Enhance Media Credibility: News publishers and media organizations can use the system to ensure their content maintains high credibility standards.[3]
3. Minimize Misinformation on Social Media Platforms: Social platforms can leverage the system to automate detection and reduce the spread of fake news across their platforms.[3]

The key objectives of the Fake News Detection System are to achieve high accuracy, ensure scalability, provide user-friendly interaction, and support real-time processing. High accuracy will be pursued through the use of advanced ML models and extensive datasets of both real and fake news, which will allow the system to learn and improve its detection capability. Scalability is essential, as the system must process a substantial volume of data daily, especially when deployed on high-traffic news and social media platforms.[4] A user-friendly interface is critical for widespread adoption, as it will encourage users of various technical backgrounds to interact with and trust the system. Real-time processing is an additional priority, ensuring users receive immediate feedback on news credibility, which is especially important for breaking news and fast-moving online content.[4]

The problem of fake news detection is challenging due to the constantly evolving nature of misinformation. New types of fake news and different misinformation strategies are continually being developed, making it necessary for the system to adapt to these changes. This system will overcome this challenge by integrating automated model updates, allowing it to stay current with trends in misinformation[3.] By using state-of-the-art machine learning methods, including Support Vector Machines, Neural Networks, and NLP-based analysis, the system can detect diverse types of fake news while adapting to new patterns.To implement this system successfully, several assumptions and dependencies are considered. Access to large, labeled datasets is essential for training and testing machine learning models[2]. These datasets must cover a wide range of topics and include both authentic and fake news examples for balanced learning. Additionally, adequate computational power and storage are required to manage and process the large datasets and conduct machine learning computations efficiently. The system will also depend on external APIs to retrieve news articles, social media posts, and updates from verified news databases. User feedback is essential for ongoing improvement, as it will allow the system to incorporate real-world input and refine its accuracy. Finally, privacy and security measures must be upheld to ensure user data is protected and to maintain trust among users.[4]

By providing a scalable, real-time, and user-friendly solution, the Fake News Detection System aims to limit the impact of misinformation and support a more informed public[5]. This document outlines the functional and non-functional requirements for building and implementing the system, describing the algorithms, system architecture, and user workflows that will enable effective and accurate detection of fake news. Through this project, we aim to contribute to the broader effort of promoting truthful and reliable information in the digital age, reducing the influence of fake news on society.[5]

3. LITERATURE SURVEY:

Deep learning for fake news detection: A comprehensive survey[Linmei Hu],[Siqi Wei]:

They showed that Deep learning (DL) methods outperform traditional machine learning (ML) methods in fake news detection, especially in handling complex features from news content, social context, and external knowledge.[5]

Fake News Detection Using Machine Learning Approaches[Z Khanam *et al* 2021] [Z Khanam[1], B N Alwasel]:The research in this paper focuses on detecting the fake news by reviewing it in two stages: characterization and disclosure. In the first stage, the basic concepts and principles of fake news are highlighted in social media. During the discovery stage, the current methods are reviewed for detection of fake news using different supervised learning algorithm.[6] Big Data ML-Based Fake News Detection Using Distributed Learning[Alaa Altheneyan]They used N-grams, HashingTF-IDF, and count vectorizer for feature extraction, followed by the suggested stacked ensemble classification model. Compared to the baseline techniques' results, the suggested model has a high classification performance of 92.45% in F1-score ensemble model improves the F1 score by 9.35%.[6]

Content-Based Fake News Detection With Machine and Deep Learning: a Systematic Review [4].For each work, the best performing algorithms and features were considered and an average performance analysis of the extrapolated data has been conducted. The outcome of the analysis shows which features and models perform better over multiple datasets.[7] More in detail, we found the most performing models to be Gradient Boosting, eXtreme Gradient Boosting, Multilayer perceptron, and Naive Bayes,

Fake news refers to deliberately fabricated news or information aimed at misleading readers. It poses significant challenges in today's digital world, impacting political, social, and economic spheres. The main challenges in detecting fake news include:Variety of

1. Content: Fake news can be in the form of articles, tweets, images, or videos.Dynamic[2]
2. New types of fake news constantly emerge.[3]
3. Mimicking Authenticity: Fake news often closely resembles legitimate news.Machine learning (ML) approaches offer promising solutions for automating the detection of such deceptive content. They can identify patterns in data to classify information as fake or truthful.[3]

4. UML DIAGRAMS:

4.1 Use case diagram

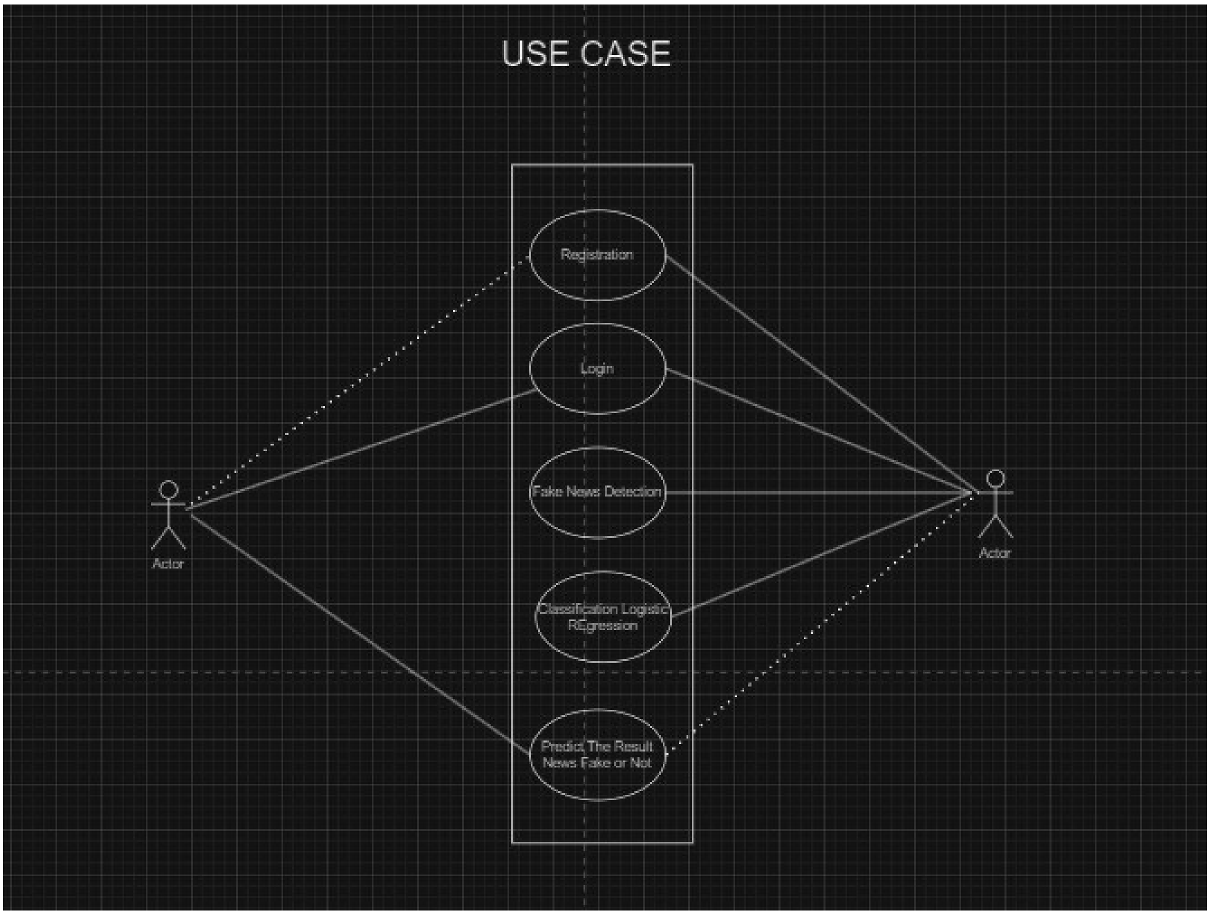

USE CASE
Registration
Login
Fake News Detection
Classification Logistic REgression
Predict The Result News Fake or Not
Actor
Actor

FIGURE 1 :Use case diagram

In fig 1 the user (represented as the "Actor") must first register within the system. During this process, the user provides necessary credentials and information to create an account.Once registered, the user logs into the system using their credentials. This step grants access to the system's core functionality, including fake news detection.After logging in, the user can utilize the system's fake news detection feature. The system allows users to input an article, which will undergo analysis to determine its authenticity.The system then predicts whether the news is fake or not based on the logistic regression output. This final prediction is communicated back to the user.

4.2 Class diagram

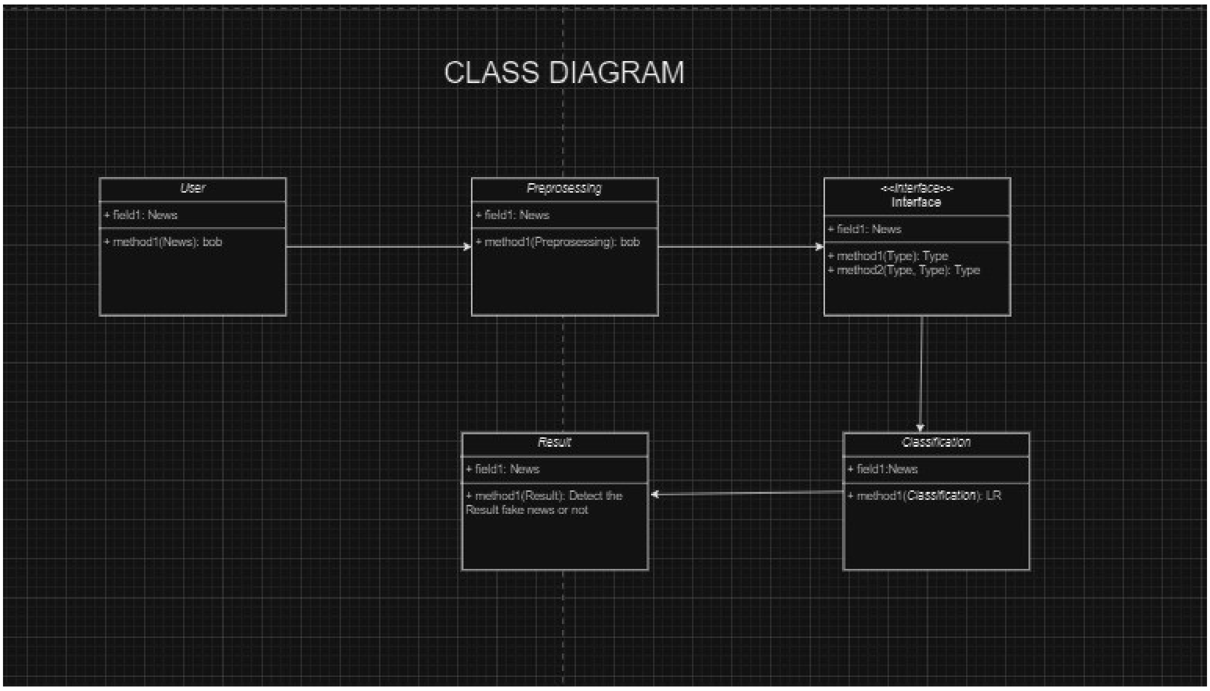


FIGURE 2:Class case diagram

Int fig 2 the User submits a news article.The article is passed to the Preprocessing class, where it undergoes necessary transformations.The preprocessed data is passed through the Interface for validation or further refinement.The processed news content is then classified in the Classification class using logistic regression.Finally, the Result class evaluates the classification outcome and determines whether the news article is fake or authentic.

4.3 Activity diagram

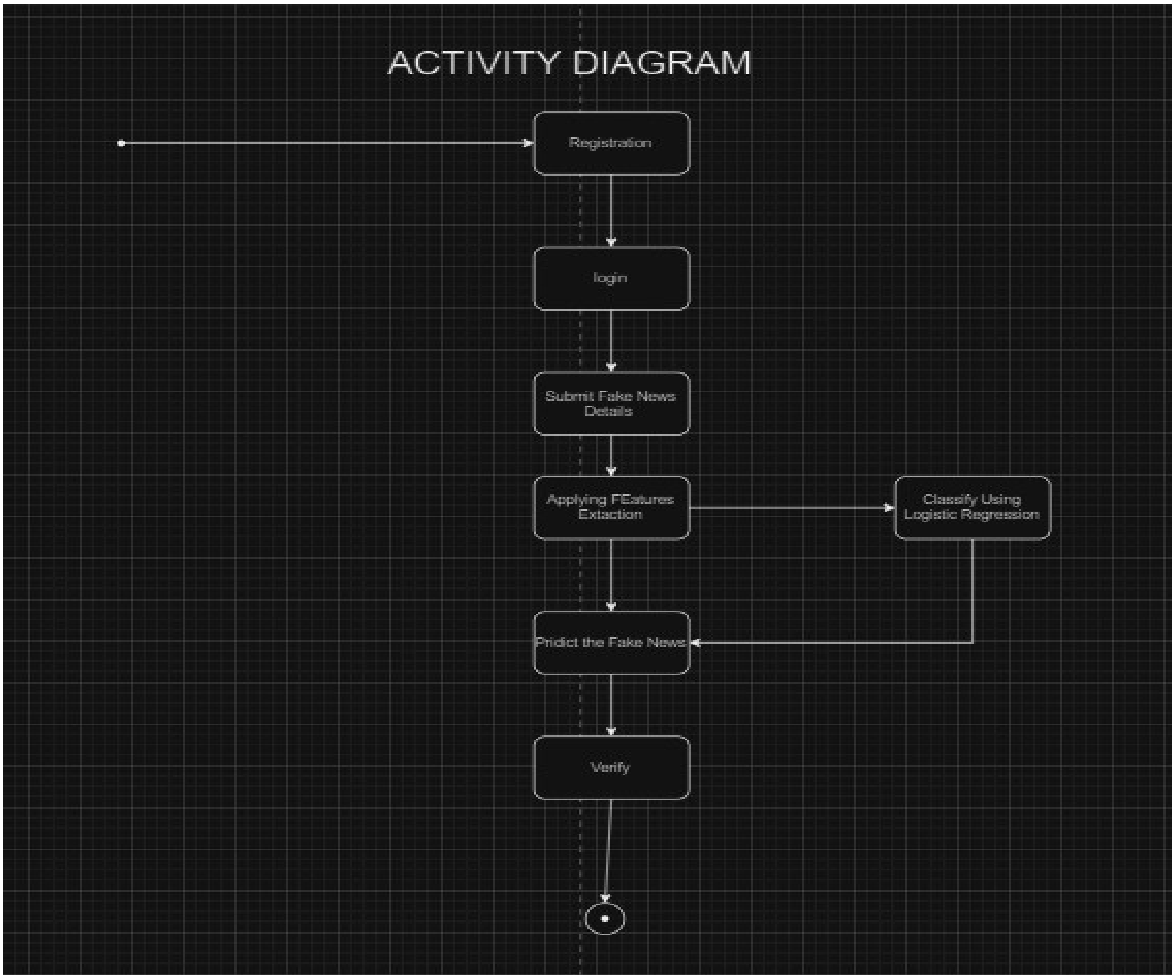


FIGURE 3:Activity diagram

In fig 3 the process begins with the user registering for an account in the system. This is a prerequisite to access the fake news detection functionality. After registration, the user must log in with their credentials to continue and use the system. Successful login grants access to the core features of the platform. Once logged in, the user submits a news article or details of the news they wish to verify for authenticity. This input is essential for the next phases of processing. The system uses the logistic regression output to predict whether the submitted news article is fake or real.

4.4 Sequence Diagram

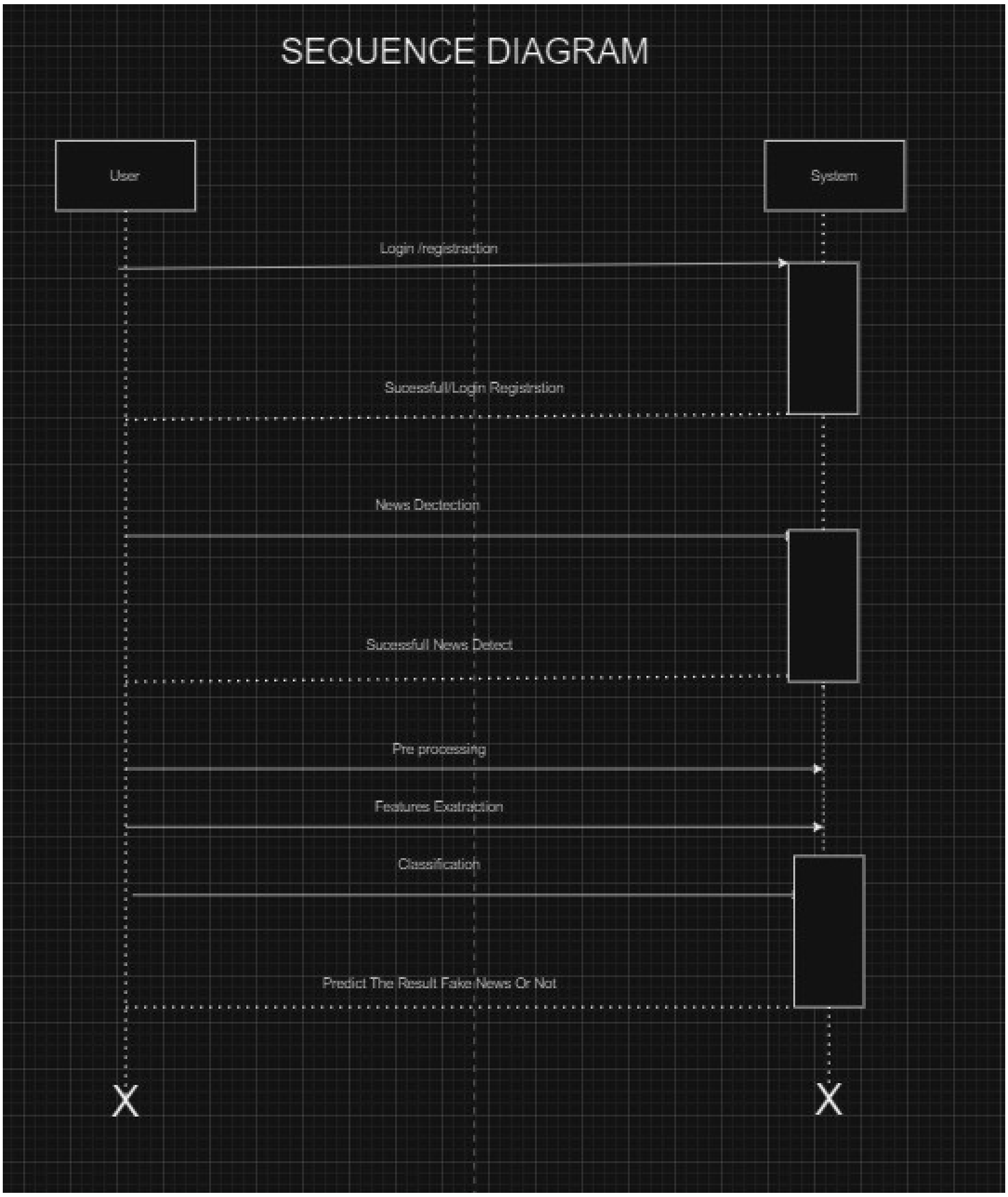


FIGURE 4 :Sequence diagram

In fig4 Actor (User) interacts with the system by sending a registration request to the System Interface.After registration, the User sends a login request with credentials to the System Interface.Once logged in, the User submits the news article or URL for fake news detection. The news content is sent to the Preprocessing.The Classification Component applies a logistic regression algorithm to classify the news as either real or fake based on the extracted features.

4.5 System Diagram

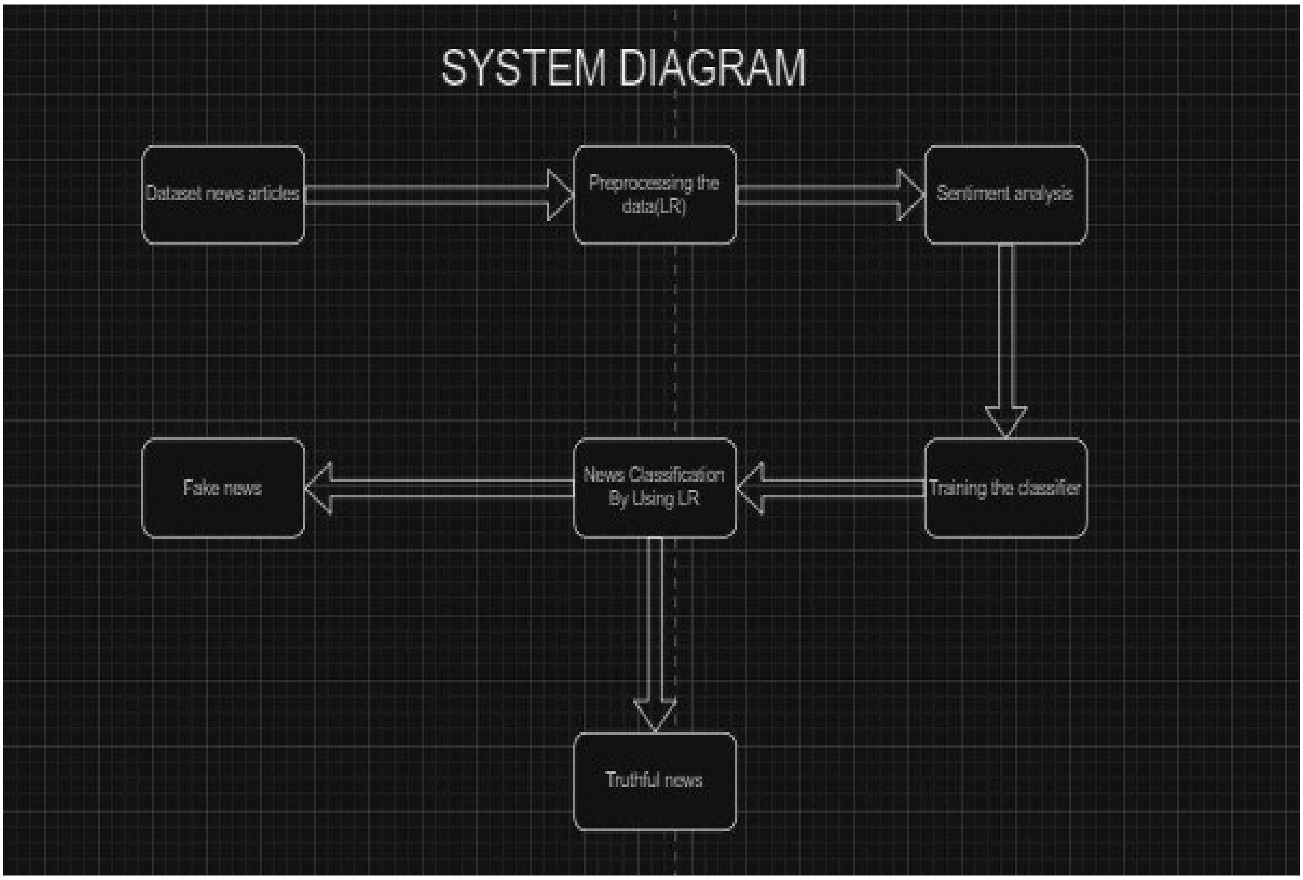


FIGURE 5 :System diagram

In fig 5 the user interacts with the system through the User Interface by submitting an article.The system preprocesses the article, extracts relevant features, and passes them to the classification model.Once a user submits a news article or URL for analysis, the content is passed to the Preprocessing ModuleThe Classification Module analyses the features using logistic regression and sends the prediction back to the user.The entire process is managed through the backend system architecture, involving various modules, databases, and machine learning components.

4.6 Data flow diagram(dfo)

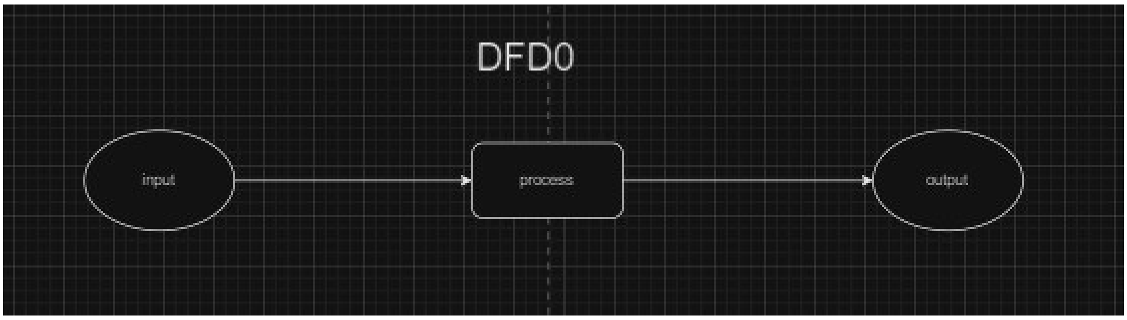


FIGURE 6 :Data flow diagram

User submits article → Preprocess → Extract Features → Classify → Return result.

4.7 DFD1 diagram

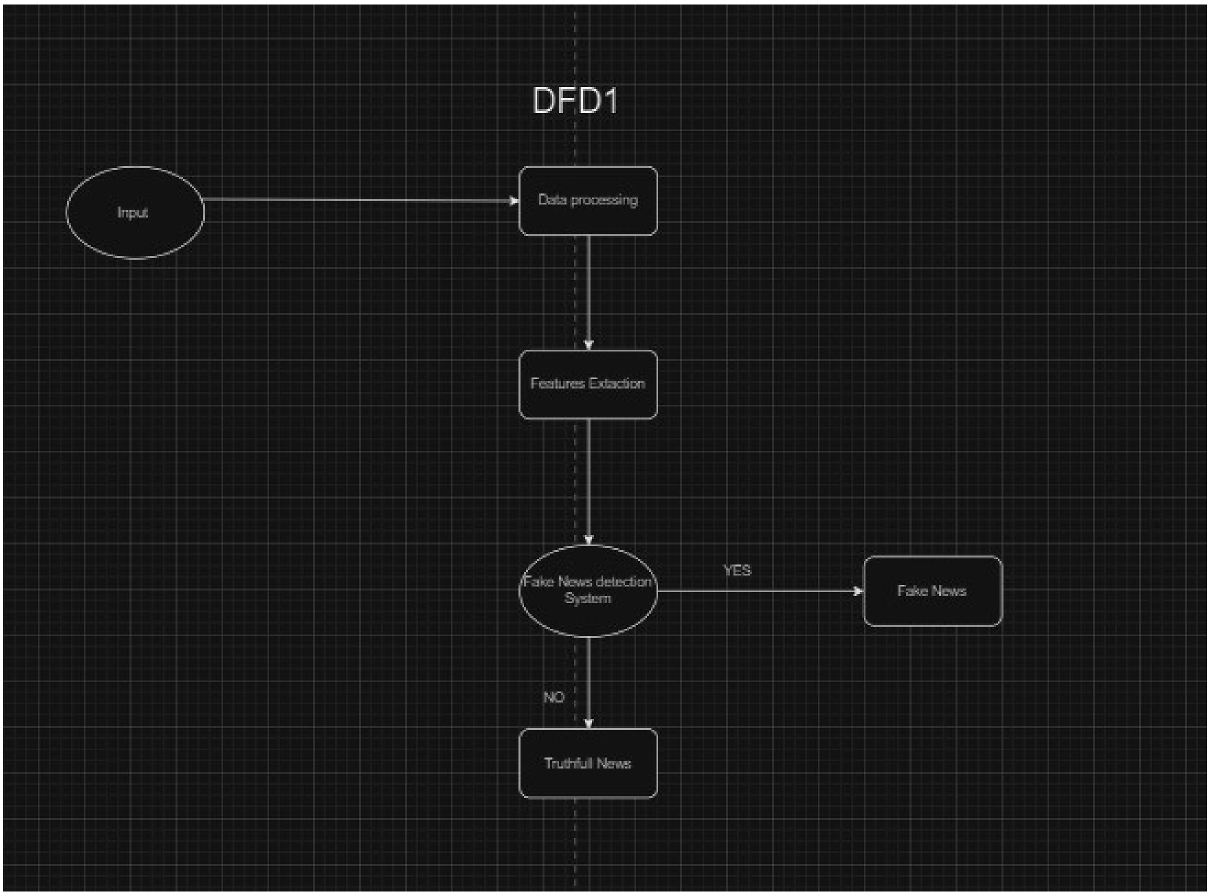


FIGURE 7::DFD1 diagram

The fig 7 describes, Input: Represents the raw data that enters the system, such as articles or news content.

Data Processing: The input undergoes a transformation or pre-processing stage, where raw data is cleaned or organized for further analysis.

Features Extraction: In this phase, significant features (such as keywords, patterns, or linguistic features) are extracted from the processed data to help in identifying fake or truthful news.

Fake News Detection System: This system analyzes the extracted features and classifies the news as either fake or truthful.YES leads to Fake News.NO leads to Truthful News.

4.8 DFD2

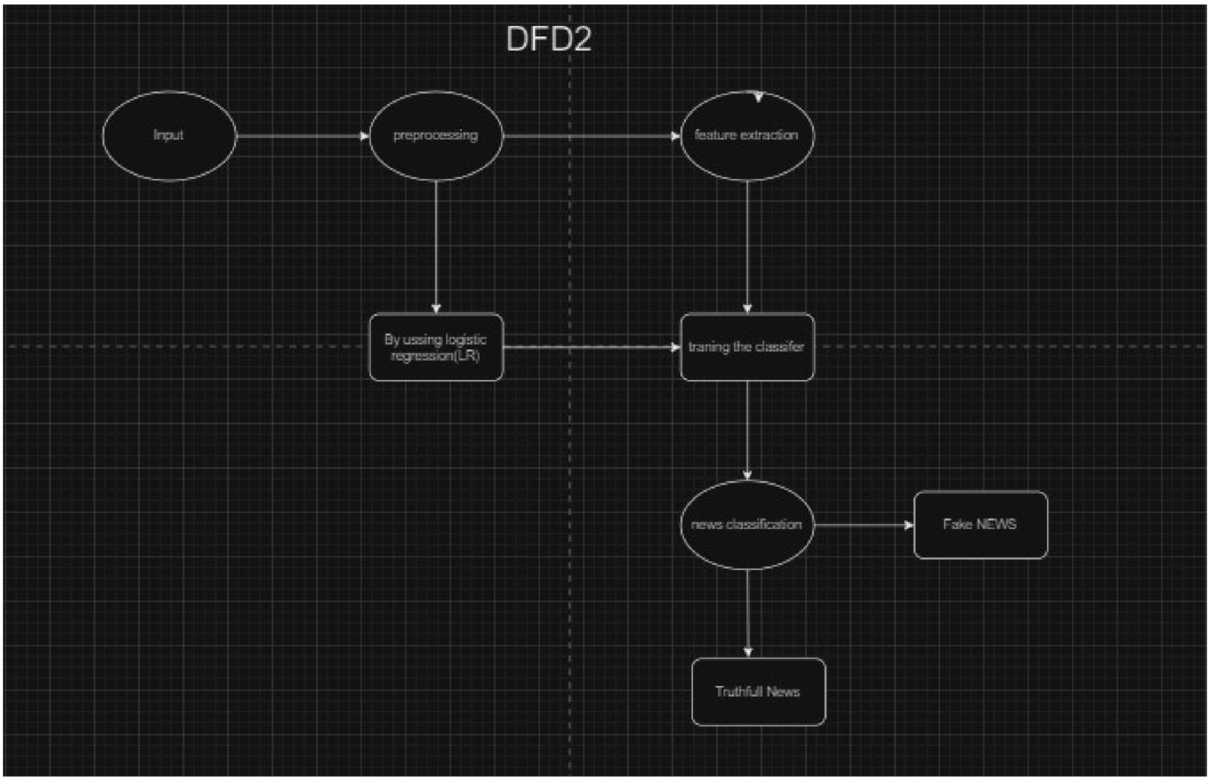


FIGURE 8 DFD2 diagram

Fig 8 states ,Input: Represents the raw news data that enters the system.

Preprocessing: The input data undergoes preprocessing, which likely involves cleaning and organizing the data for further steps (e.g., removing noise, normalizing text)..

By Using Logistic Regression (LR): The extracted features are used in a logistic regression model for classification. Logistic regression is a machine learning algorithm often used for binary classification tasks like determining whether news is fake or truthful.

Training the Classifier: The logistic regression model is trained using the features, which helps the system learn from past data to improve the accuracy of future predictions and classify them into fake and true news

4.9 ER diagram

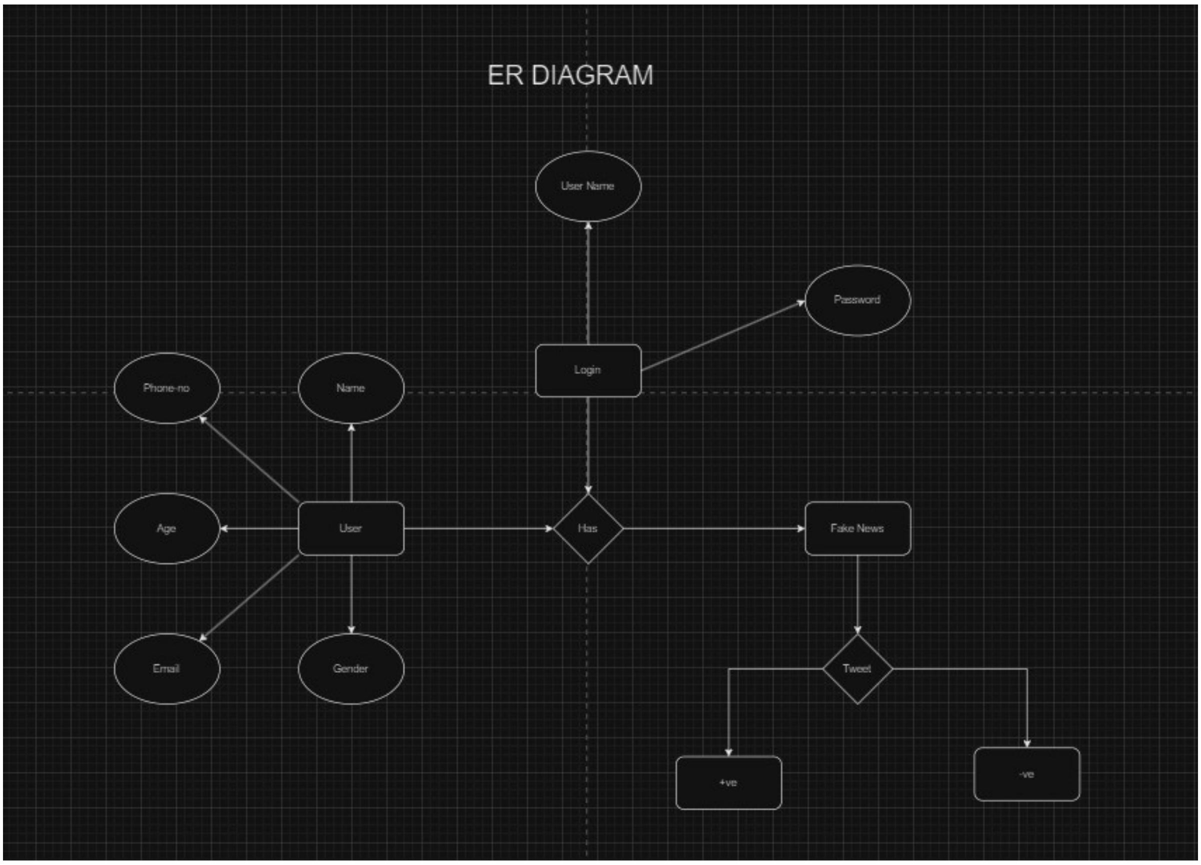


FIGURE 9 :ER diagram

Fig 9 captures the relationships between users, login credentials, fake news, and user-generated tweets, illustrating the interactions within a system that deals with fake news detection and user engagement.

5. FUNCTIONAL AND NON FUNCTIONAL REQUIREMENTS

Functional Requirements:

1. **Content Ingestion and Preprocessing**: The system must accept input data in the form of news articles, social media posts, or URLs. It should preprocess the input text by cleaning, tokenizing, and normalizing the data for consistent analysis.
2. **Text Analysis and Feature Extraction:** The system must analyse text to extract linguistic, stylistic, and semantic features that contribute to the classification of fake news..
3. **Machine Learning Model for Classification:** The system must implement machine learning models to classify content as “fake” or “real.”

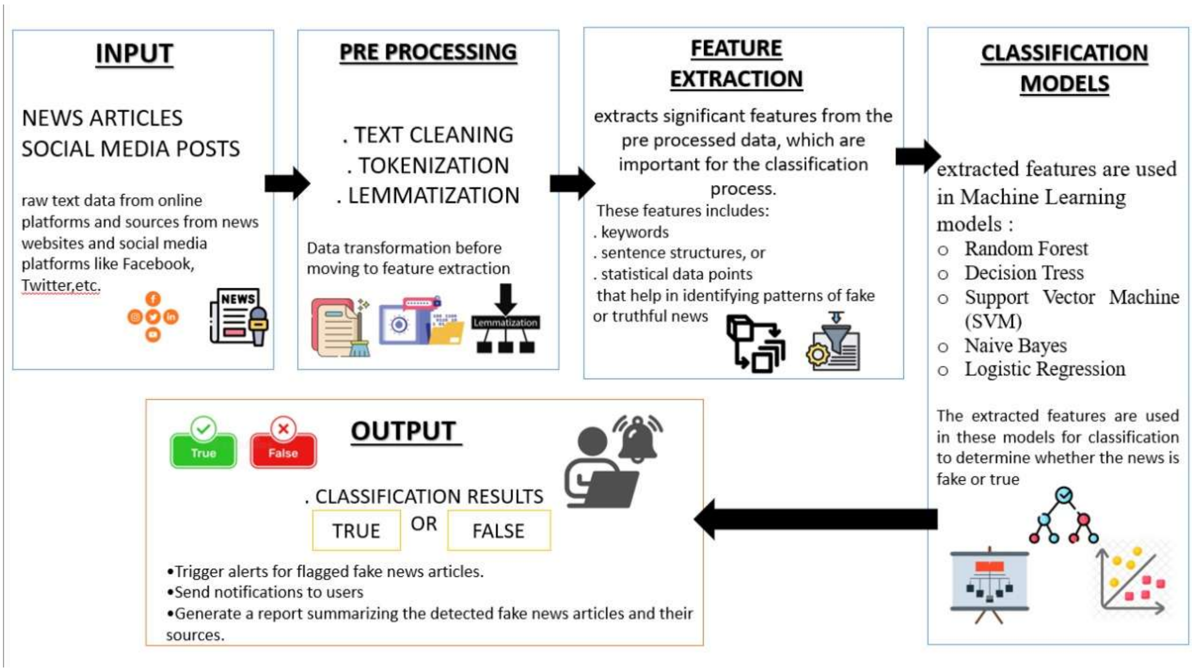
4. **Source Verification:** The system should cross-check the credibility of sources mentioned in the content by comparing them with a list of known reputable and non-reputable sources .If an unverified source is detected, the system should factor this into its classification score.
5. **Scoring and Confidence Level:** The system must assign a credibility score (e.g., 0-100%) indicating the likelihood of content being fake. A threshold value should be set to label content as “fake” or “real” based on the score, which can be adjusted by administrators.
6. **User Feedback Mechanism** The system should provide a feedback option for users to flag content they believe was incorrectly classified. This feedback should be collected and used to retrain and improve the system's accuracy over time.
7. **Real-Time Processing:** The system must be able to analyse and classify news articles in real-time or near real-time to provide timely results. It should notify users promptly once the analysis and classification are complete.
8. **Result Presentation and Explanation:** The system should display the classification results, including the credibility score and a summary of key features contributing to the classification. Users should be able to view an explanation of why the content was classified as “fake” or “real,” including specific linguistic markers and source reliability.
9. **Administrator Controls:** Administrators should have access to controls for adjusting the classification thresholds, retraining the model, and managing source credibility lists.
10. **Database Management:** The system must maintain a database for storing analysed articles, results, flagged content, and user feedback. It should support easy retrieval of past analyses for auditing purposes.

Non- Functional Requirements:

1. **Performance and Scalability**: The system must be able to handle large volumes of incoming data .It should scale efficiently to accommodate spikes in usage, such as during high-traffic events.
2. **Accuracy and Reliability:** The machine learning models should be highly accurate, with a minimum target accuracy of 90% on classification. The system should maintain consistent accuracy across various domains and adapt to new patterns of misinformation.
3. **Usability:** The system must provide a user-friendly interface, making it accessible to users with varying levels of technical expertise. The explanation of results should be clear and easy to understand, even for non-technical users.
4. **Adaptability**: The system should be flexible enough to integrate new data sources and model updates as new types of fake news and misinformation trends emerge. It should support retraining capabilities to improve performance based on updated datasets and user feedback.
5. **Security and Privacy**: The system must ensure secure handling of user data, following data privacy standards and regulations.
6. **Real-Time Responsiveness**: The system should complete its analysis within a few seconds for smaller posts and up to a few minutes for longer articles, ensuring users receive timely feedback.
7. **Maintainability and Extensibility**: The codebase should be well-organized and documented, allowing easy updates, troubleshooting, and extension of features. New machine learning models or modules should be easily integrable as the technology evolves.
8. **Availability and Reliability**: The system must ensure high availability, with minimal downtime, to support users continuously.
9. **Compliance**: The system should comply with relevant regulations, such as data protection laws (e.g., GDPR), ensuring user data is secure and usage is transparent.
10. **Logging and Monitoring**: The system must log important events, errors, and user feedback, supporting tracking and troubleshooting.

6 PROPOSED SYSTEM

6.1 Block diagram:



INPUT

NEWS ARTICLES
SOCIAL MEDIA POSTS

raw text data from online platforms and sources from news websites and social media platforms like Facebook, Twitter,etc.

PRE PROCESSING

. TEXT CLEANING
. TOKENIZATION
. LEMMATIZATION

Data transformation before moving to feature extraction

FEATURE EXTRACTION

extracts significant features from the pre processed data, which are important for the classification process.
These features includes:
. keywords
. sentence structures, or
. statistical data points
that help in identifying patterns of fake or truthful news

CLASSIFICATION MODELS

extracted features are used in Machine Learning models :

- Random Forest
- Decision Tress
- Support Vector Machine (SVM)
- Naive Bayes
- Logistic Regression

The extracted features are used in these models for classification to determine whether the news is fake or true

OUTPUT

. CLASSIFICATION RESULTS
TRUE OR FALSE

•Trigger alerts for flagged fake news articles.
•Send notifications to users
•Generate a report summarizing the detected fake news articles and their sources.

Figure 10:data block diagram

Figure 10 outlines the process of a Fake News Detection System using machine learning, from input data to classification results. The system starts by collecting raw text data from news articles and social media posts, sourced from platforms like Facebook and Twitter. This text undergoes preprocessing, where it is cleaned, tokenized, and lemmatized to ensure consistency and standardization before moving forward.

Once the data is pre-processed, significant features are extracted to facilitate the classification process. These features include keywords, sentence structures, and various statistical indicators that help identify patterns commonly associated with either fake or authentic news. The extracted features are then used as inputs for machine learning models such as Random Forest, Decision Trees, Support Vector Machines, Naive Bayes, and Logistic Regression, which collectively help classify the content.

Finally, the system provides an output that categorizes the content as either "True" (authentic) or "False" (fake), along with additional actions like triggering alerts, sending notifications to users, and generating summary reports. This output is designed to assist users in identifying and managing fake news effectively.

The Fake News Detection System is a comprehensive software solution developed to address the challenge of misinformation in digital media. This system uses machine learning (ML) algorithms and natural language processing (NLP) techniques to analyze news articles and social media posts, identifying patterns commonly associated with fake news. By automating the detection process, the system provides users with a quick and accurate assessment of content credibility, reducing the influence of false information and supporting more informed decision-making.

6.2 PSEUDOCODES:

Decision Tree Pseudo-code-

```
 GenerateDecisionTree(Sample s, features F)
 1. If stop _conditions(S,F) = true then
        a. leaf = create_Node()
        b. Leaf.lable= classify(s)
        c. Return leaf
2. root = create_Node()
3. root.testcondition = find_bestSplit(s,f)
4. v = { v l v a possible outcome of root.testconditions)
5. for each value v∈ V:
6. sv: = {s∣root.testcondition(s) = v and s∈ S};
7. child = Tree_Growth(Sv ,F) ;
8. Grow child as a descent of roof and label the edge (root→child) as v
         Return root
```

Random Forest Pseudo-Codez-

```
    To generate c classifiers:
   For i=1 to c do
   Randomly sample the training data with replacement to produce Di
   Create a root node Ni containing Di
   Call BuildTree(Ni)
   End For
    BuildTree(N):
   If N contain instances of only one class then
       Return
   Else
       Randomly select x% of the possible splitting features in N
       Select the features F with the Highest information gain to split on
       Create f child nodes of Ni….Nf where F has f possible values.
       For i=0 to f do
       Set the contents of Ni to Di where Di is all instances inN that match Fi
       Call BuildTree(Ni)
       End for
   End if
```

SVM Pseudo-Code

```
F[0..N-1]: a feature set with N features that is sorted by information gain in decreasing order accuracy(i): accuracy of a prediction model based on SVM with F[0...i] gone set
low = 0
high = N-1
value = accuracy(N-1)
IG_RFE_SVM(F[0...N-1], value, low, high) {
    If (high } > low)
         Return F[0...N-1] and value
        mid = (low + high ) / 2
         value_2 = accuracy(mid)
    If (value_2 < value)
        return IG_RFE_SVM(F[0...mid], value_2, low, mid)
    Else (value_2 > value)
            return IG_REF_SVM(F[0...high], value, mid, high)
```

Naïve Bayes algorithm

1. Read Training Dataset T;
2. Calculate the mean and norm of each class's predictor variables;
3. Repeat
4. Calculating the likelihood of using the equation of gauss density in each class;
5. Until Pending the estimation of the likelihood of all predictor variables (f1, f2, f3,..., fn),.
6. Calculated the likelihood for respective class;
7. Get the highest likelihood

Logistic Regression Pseudocode

1. Initialize weights (W) and bias (b) to small random values

```
2.  Set learning rate ($\alpha$) and number of iterations (num_iterations)
3.  For i = 1 to num_iterations:
        a.  Compute the linear combination Z: $Z = W * X + b$
        b.  Apply the sigmoid function to get the predicted probability:  $P = 1 / (1 + \exp(-Z))$
        c.  Compute the error (loss) using log-loss function:
4.  $Loss = -(1/m) * \Sigma [y * \log(P) + (1 - y) * \log(1 - P)]$
        a.  Compute gradients for weights and bias:
5.  $dW = (1/m) * \Sigma [(P - y) * X]$
6.  $db = (1/m) * \Sigma (P - y)$
        a.  Update weights and bias using gradient descent:
7.  $W = W - \alpha * dW$
8.  $b = b - \alpha * db$
9.  Return the learned weights (W) and bias (b)
10. For prediction (new data point X_new):
        a.  Compute $Z\_new = W * X\_new + b$
        b.  Compute probability $P\_new = 1 / (1 + \exp(-Z\_new))$
        c.  If $P\_new > 0.5$, classify as class 1 (e.g., fake news)
11. Else, classify as class 0
```

6.3 FLOWCHART

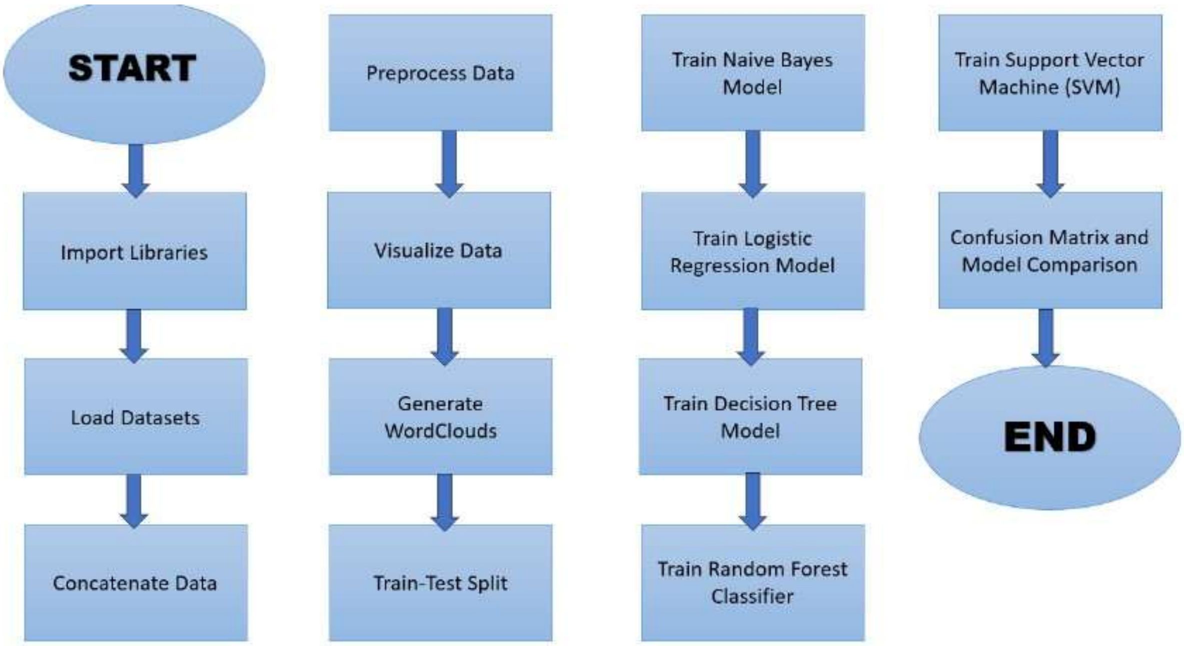

START
Import Libraries
Load Datasets
Concatenate Data
Preprocess Data
Visualize Data
Generate WordClouds
Train-Test Split
Train Naive Bayes Model
Train Logistic Regression Model
Train Decision Tree Model
Train Random Forest Classifier
Train Support Vector Machine (SVM)
Confusion Matrix and Model Comparison
END

Figure 10 Flowchart

7 RESULTS:

The scope of this project is to tell if the news data, of a dataset known as TRUE and FALSE, labeled by fake or trust news. We have performed analysis on this dataset . The results of the analysis of the datasets using the six algorithms have been depicted using the confusion matrix. Thefive algorithms used for the detection are as:

- Logistic Regression.
- Random Forests.

- Naive Bayes.
- Decision Tree.
- SVM

The confusion matrix is automatically obtained by Python code using the cognitive learning library when running the algorithm code in Anaconda platform. The Confusion Matrix for all the algorithms are depicted below

7.1. LOGISTIC REGRESSION CONFUSION MATRIX

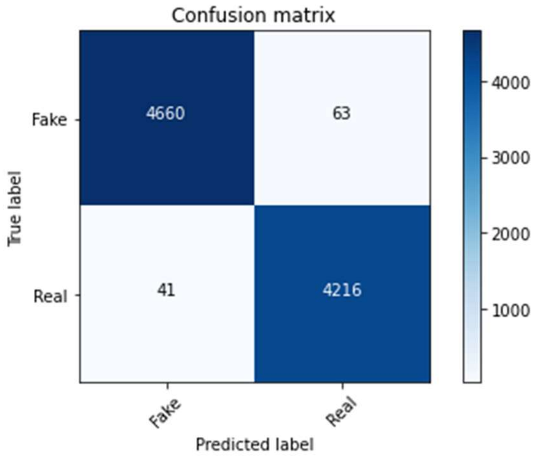


Figure: 11 LR confusion matrix

7.2 NAÏVE BAYES CONFUSION MATRIX

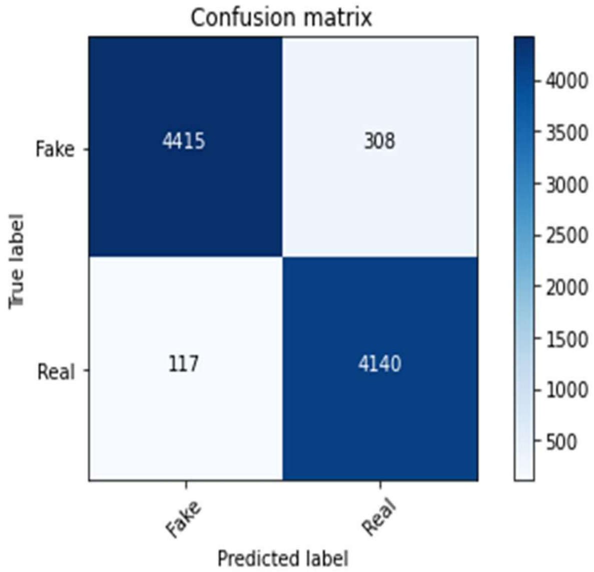


Figure:12 NAÏVE BAYES CONFUSION MATRIX

7.3 DECISION TREE CONFUSION MATRIX

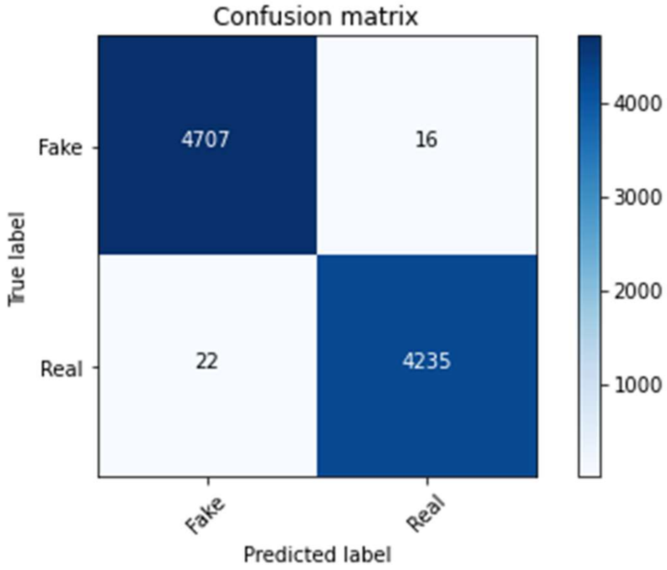


Figure 13 decision tree confusion matrix

7.4 RANDOM FOREST CONFUSION MATRIX

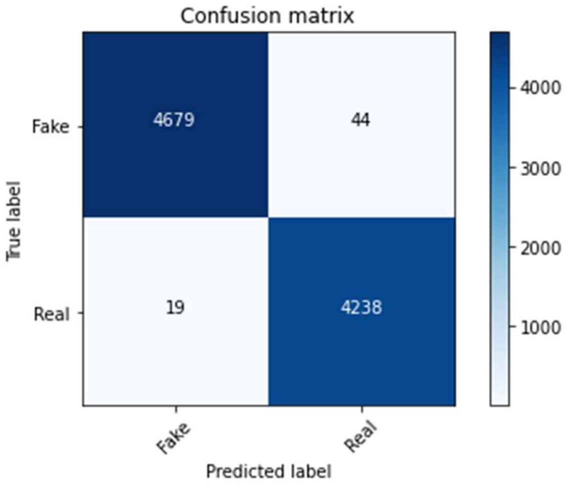


Fig:14 Random forest CONFUSION MATRIX

7.5 SVM CONFUSION MATRIX

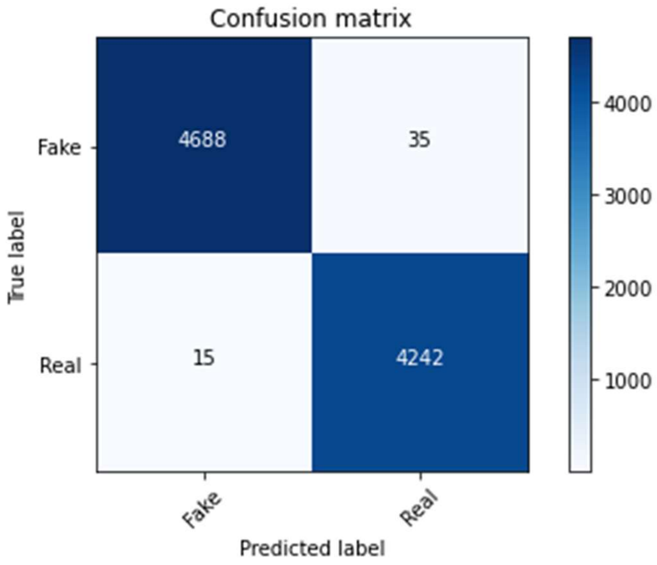


Fig:15 SVM confusion matrix

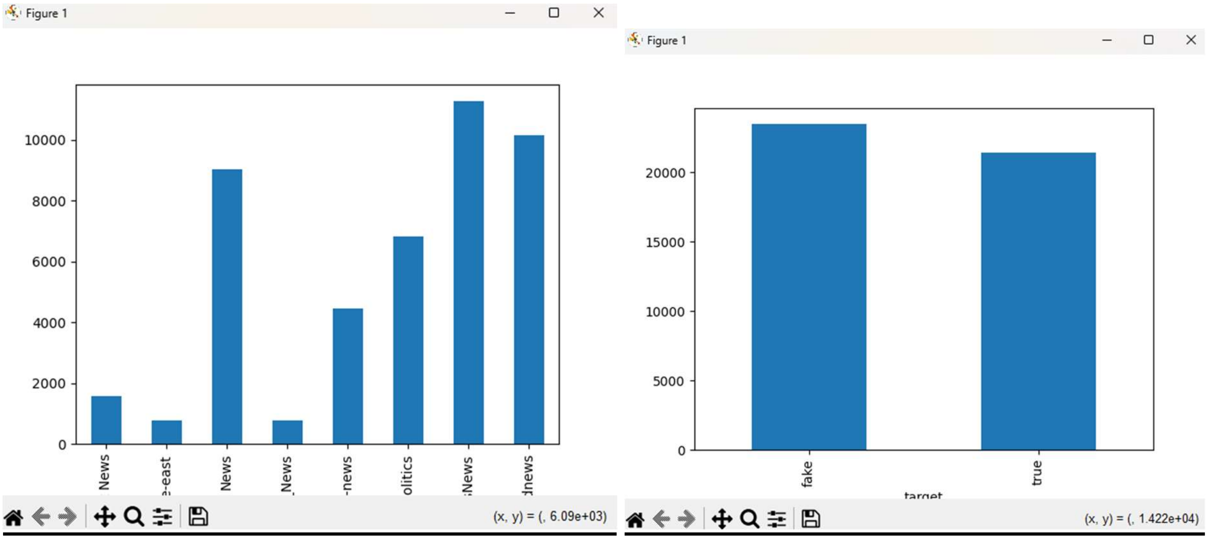


Figure:16 Bar graph of results

Figure 16 represents the comparisons between the models and sates the fake and actual news

TABLE 1

Text	Subject	Target
nairobi reuters south sudan hiking fees humani..	World news	True
reuters alabama governor robert bentley said t…..	Politics news	True
media trying best control see see comes alltelf...	Politics news	fake
dublin reuters british prime minister Theresa….	Worldnews	True
footage georgia restaurant worker taking time ...	news	fake

TABLE 1:News channel news True or false

TABLE 1 provides a dataset sample for a fake news detection system. It contains three columns: Text, Subject, and Target.The Text column includes short excerpts or phrases from news articles. These samples contain keywords such as "nairobi reuters," "alabama governor," and "georgia restaurant," indicating the location or context of the news.The Subject column categorizes the general topic of each text sample, such as "World news," "Politics news," or simply "news." These categories help in identifying the type of content and its relevance to the classification process.

The Target column shows the truthfulness of each text sample, marked as either "True" or "fake." This label indicates whether the content is authentic or false, which serves as the ground truth for training and evaluating the machine learning models.The table contains five sample entries, with a mix of true and fake news labels across different subjects. This format is typically used in supervised machine learning for training a model to classify news articles as either true or fake.

TABLE 2

SUBJECT	COUNT
Govt news	1570
Middile east	778
News	9050
Us news	783
Left news	4459
Politics	6841
Politics news	11272
worldnews	10145

TABLE 2:Count of fake news per channel

This table, labeled "TABLE 2," presents data on various news channels and the number of fake news incidents attributed to each. The columns are titled "SUBJECT" (for the name of the news channel) and "COUNT" (for the count of fake news incidents). The channels with the highest numbers of fake news incidents are "Politics news" (11,272), "World news" (10,145), and "News" (9,050). This suggests these sources have the most reported instances of fake news.

8. Conclusion

Our fake news detection model, built using machine learning, demonstrates an effective framework for automating the identification of fake news, offering a scalable and efficient approach to managing the overwhelming volume of information on digital platforms.[4]

The models are trained and tested on a balanced dataset of fake and truthful news. The pipeline evaluates the performance of each model using accuracy metrics and confusion matrices. Overall, the model results show high accuracy across multiple classifiers, with logistic regression, SVM, and random forest achieving notable performance.[3] However, there are limitations in terms of bias, context understanding, and the ability to generalize across domains, highlighting areas for further improvement and exploration in future work.[2]

By defining clear functional and non-functional requirements, this document ensures that the system is both efficient and scalable, capable of handling large data volumes, and adaptable to evolving misinformation trends. The combination of real-time processing, user-friendly interfaces, and robust classification models aims to deliver accurate and reliable results to users, empowering them to make informed decisions regarding the credibility of the information they encounter.

The model demonstrates the potential of machine learning in automating the process of fake news detection, but care must be taken in addressing the limitations to ensure the system's robustness in real-world applications.

9. References:

1 . *IOP Conf. Ser.: Mater. Sci. Eng.* 1099 012040 , IOP Conference Series: Materials Science and Engineering Z Khanam[1], B N Alwasel[1], H Sirafi[1] and M Rashid ,https://iopscience.iop.org/article/10.1088/1757-899X/1099/1/012040/meta

2. E. C. Tandoc Jr et al. “Defining fake news a typology of scholarly definitions”. Digital Journalism , 1–17. 2017

3. Iftikhar Ahmad, Muhammad Yousaf, Suhail Yousaf, Muhammad Ovais Ahmad , Fake News Detection Using Machine Learning Ensemble Methods

4. Rajat Subhra Goswami , Lilapati Waikhom, *Proceedings of International Conference on Advancements in Computing & Management (ICACM) 2019*

5. OP Conference Series: Materials Science and Engineering https://iopscience.iop.org/article/10.1088/1757-899X/1099/1/012040

6. Hoang Nguyen https://www.researchgate.net/figure/The-ANN-5-8-8-1-model-for-estimating-MCC-in-this-study_fig4_335375894

7. https://www.scribbr.com/category/research-paper/

8. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10078408 Z Khanam[1], B N Alwasel[1], H Sirafi[1] and M Rashid[2]