D Y PATIL
INTERNATIONAL
UNIVERSITY
AKURDI PUNE

Project Report

On

Fake News Detection

Submitted to D Y Patil International University, Akurdi, Pune in partial fulfilment of full-time degree

Master of Computer Applications

Submitted By:

Name: Prerna Maheshbhai Patil

PRN: 20240804063

Under the Guidance of

(Ms. Asha Ayakar)

School of Computer Science, Engineering and Applications

D Y Patil International University, Akurdi,Pune, INDIA, 411044

[Session 2024-2025]

D Y PATIL
INTERNATIONAL
UNIVERSITY
AKURDI PUNE

CERTIFICATE

This is to certify that the work entitled “Fake News Detection” submitted as project II is a bonafide work carried out by Prerna Maheshbhai Patil in partial fulfillment of the award of the degree of Master of Computer Applications , D Y Patil International University, Pune, during the academic year 2024- 2025. The project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the Master of Computer Applications.

Ms. Asha Ayakar
(Project Guide)

Dr. Swapnil Waghmare
(Project Coordinator)

Dr. Maheshwari Biradar
(HOD, BCA & MCA)

Dr. Rahul Sharma
(Director)

School of Computer Science Engineering & Applications
D Y Patil International University, Akurdi
Pune, 411044, Maharashtra, INDIA

DECLARATION

I, hereby declare that the following Project entitled Fake News Detection is an authentic documentation of my own original work to the best of my knowledge. The following Project and its report in part or whole, has not been presented or submitted by me for any purpose in any other institute or organization. Any contribution made to my work, with whom I have worked at D Y Patil International University, Akurdi, Pune, is explicitly acknowledged in the report.

Name: Prerna Maheshbhai Patil
PRN No: 20240804063 Signature :

ACKNOWLEDGEMENT

With due respect, I express my deep sense of gratitude to respected guide Ms. Asha Ayakar, for her valuable help and guidance. I am thankful for the encouragement that she has given us in completing this Project successfully.

It is imperative for me to mention the fact that the report of project could not have been accomplished without the periodic suggestions and advice of our project supervisor (Name).

I am also grateful to our respected, Dr. Rahul Sharma(Director), Dr. Maheshwari Biradar (HOD, BCA & MCA) and (Hon'ble Vice Chancellor, DYPIU, Akurdi) Prof. Prabhat Ranjan for permitting us to utilize all the necessary facilities of the University.

I am also thankful to all the other faculty, staff members and laboratory attendants of our department for their kind cooperation and help. Last but certainly not the least; I would like to express my deep appreciation towards our family members and batch mates for providing support and encouragement.

Name: Prerna Maheshbhai Patil
PRN: 20240804063

Abstract

Fake news has emerged as a major threat in today's digital era, where misinformation spreads rapidly across online platforms, influencing public opinion and creating social unrest. Manual verification is time-consuming and insufficient at scale, making the development of automated fake news detection systems essential. This project leverages Natural Language Processing (NLP) and Machine Learning (ML) techniques to classify news articles as real or fake, offering a reliable and scalable solution to combat misinformation.

The project begins with data preprocessing, including text cleaning, stopword removal, tokenization, lemmatization, and text preprocessing. Exploratory Data Analysis (EDA) helps identify trends in subject distribution, text length variation, and common word usage via WordClouds. Feature extraction is performed using TF-IDF vectorization along with linguistic attributes like word count, character count, sentiment score, readability score, POS tagging, and Named Entity Recognition (NER).

Several ML models are implemented: PassiveAggressiveClassifier, Naïve Bayes, Logistic Regression, Decision Tree, Random Forest, and an Ensemble Voting Classifier. Models are evaluated using accuracy, precision, recall, and F1-score, achieving over 95% accuracy without requiring hyperparameter tuning. To enhance model transparency, LIME (Local Interpretable Model-agnostic Explanations) is used to highlight the most influential features in each prediction.

By integrating advanced NLP techniques, diverse ML models, and explainable AI tools, this project delivers a robust fake news detection system. It supports real-time verification, assists users in identifying misleading content, and contributes toward promoting trustworthy digital information.

List of Figures

TABLE OF CONTENTS

1. INTRODUCTION

1.1. Background

Fake news refers to intentionally false or misleading information that spreads rapidly through websites, social media, and other digital platforms. It can distort facts, manipulate public opinion, and contribute to social and political instability. With the high speed and volume of online information, identifying fake news has become increasingly challenging.

To address this issue, automated fake news detection systems use Machine Learning (ML) and Natural Language Processing (NLP) techniques to analyze news content. These systems evaluate text patterns, language features, and linguistic cues to classify articles as real or fake. Algorithms such as PassiveAggressiveClassifier, Naïve Bayes, Random Forest, and Ensemble models are used to improve prediction accuracy.

This project aims to build a reliable and scalable fake news detection system that supports real-time verification and helps reduce the spread of misinformation. By combining ML, NLP, and explainable AI techniques like LIME, the system enables users to access more trustworthy digital content.

1.2. Objectives

The primary objective of this project is to develop an automated system for detecting fake news using Machine Learning (ML) and Natural Language Processing (NLP) techniques. The specific objectives are as follows:

- To collect and preprocess news data by performing cleaning, stopword removal, lemmatization, and other NLP-based transformations.
- To conduct Exploratory Data Analysis (EDA) to identify patterns in fake and real news, including subject distribution, word usage, and text length.
- To extract relevant features using techniques such as TF-IDF vectorization, sentiment scoring, readability scores, POS tagging, and NER.
- To train and evaluate multiple ML models including PassiveAggressiveClassifier, Naïve Bayes, Logistic Regression, Decision Tree, Random Forest, and Ensemble Voting Classifier.

- To measure the performance of models using metrics like accuracy, precision, recall, and F1-score.
- To implement LIME (Local Interpretable Model-Agnostic Explanations) for interpreting and explaining individual predictions made by the model.
- To provide a reliable and scalable system that helps in real-time detection of fake news and reduces the spread of misinformation online.

1.3. Purpose

The purpose of this project is to develop an intelligent system capable of detecting fake news content circulating on digital platforms. With the increasing reliance on online news sources, it has become essential to filter out false or misleading information that can negatively influence public perception and decision-making.

This project aims to use Machine Learning (ML) and Natural Language Processing (NLP) techniques to automatically classify news articles as real or fake. By analyzing linguistic patterns and applying effective classification models, the system will help in fact-checking, promote reliable information, and minimize the spread of misinformation.

Ultimately, the goal is to create a robust, scalable, and interpretable fake news detection solution that can be integrated into real-time environments, offering users a dependable tool for validating digital content.

1.4. Scope

This project aims to build an automated fake news detection system using Machine Learning and Natural Language Processing techniques. It covers the entire process from data preprocessing and feature extraction to model training, evaluation, and interpretation. The focus is on developing an accurate, interpretable, and efficient system for classifying news as real or fake.

- Preprocessing of textual news data using techniques such as text cleaning, stopword removal, and lemmatization.
- Extraction of meaningful features including TF-IDF, sentiment scores, readability metrics, word/character count, POS tagging, and NER.
- Exploratory Data Analysis (EDA) to identify patterns in subject categories, word usage, and text length distribution.

- Implementation of multiple ML models: PassiveAggressiveClassifier, Naïve Bayes, Logistic Regression, Decision Tree, Random Forest, and Ensemble Voting Classifier.
- Evaluation of model performance using accuracy, precision, recall, and F1-score.
- Integration of LIME (Local Interpretable Model-Agnostic Explanations) to visualize and interpret the predictions made by the models.

1.5. Applicability

The machine learning system developed in this project can be applied by media companies, social media platforms, and news organizations to:

- Detect fake news and misinformation before it spreads widely.
- Take immediate action to flag or remove misleading news articles from public view.
- Improve content verification processes by leveraging data-driven insights.
- Save time and resources by automating the process of identifying suspicious news sources.
- Make better decisions regarding content publication and distribution using machine learning and analytics.

This system can also be useful for data analysts, journalists, policymakers, and fact-checking organizations in other sectors facing misinformation challenges, such as healthcare, politics, and education, with slight adjustments to the dataset.

2. PROJECT PLAN

2.1. Problem Statement

In today's digital world, the spread of fake news is a major challenge, especially with the vast amount of information shared through social media and online news platforms. Fake news, which often comes from unreliable sources, can easily influence public opinion and disrupt social and political stability. On the other hand, real news, particularly from trusted sources related to world and politics, is often overshadowed by misleading information. Traditional methods of detecting fake news are slow, manual, and fail to address the scale of the problem.

There is a need for an automated system that can efficiently distinguish between real and fake news. By applying machine learning techniques, this system can analyze news articles and identify patterns that differentiate between credible and false content. This predictive system can help media organizations and social media platforms take proactive steps to curb the spread of misinformation.

The main problem this project addresses is how to accurately detect fake news using machine learning models trained on news data, helping organizations improve content credibility verification and ensure that only reliable information, especially related to world and political news, reaches the public.

2.2. Requirement Specification

Functional Requirements

- The system should accept news articles as input, either manually or via CSV upload.
- The system should preprocess the input data, including text cleaning, tokenization, and removing stop words.
- The system should handle class imbalance using techniques like oversampling or undersampling.
- The system should apply multiple machine learning models (e.g., Logistic Regression, Random Forest, SVM) to classify news articles as real or fake.
- The system should evaluate model performance using metrics such as accuracy, precision, recall, and F1-score.

- The system should provide real-time predictions for news articles, indicating whether they are real or fake.
- The system should display prediction results with confidence levels and an explanation (e.g., LIME).
- The system should allow batch predictions through uploaded CSV files containing multiple news articles.
- The system should maintain logs for each prediction and provide explanations for model decisions.

Non-Functional Requirements

- The system should be user-friendly and easy to navigate, with a simple interface for input and results display.
- The system should provide quick response times for real-time predictions.
- The system should ensure data security and user privacy when processing user data.
- The system should be scalable to handle large volumes of news articles if required.
- The system should maintain model interpretability and transparency, offering insights into why a news article is classified as real or fake.

Software Requirements

- Operating System: Windows / Linux
- Programming Language: Python 3.9+
- Libraries: pandas, numpy, scikit-learn, matplotlib, seaborn, xgboost, imbalanced-learn, nltk, lime
- Web Framework: Streamlit
- IDE: Jupyter Notebook / VS Code

Hardware Requirements

Minimum Requirements

- Processor: Intel Core i3 (8th Gen) or equivalent

- RAM: 4 GB
- Storage: 5 GB of free disk space

Recommended Requirements

- Processor: Intel Core i5 (10th Gen or higher) / AMD Ryzen 5 or better
- RAM: 8 GB or higher
- Storage: 10 GB or more of free disk space (preferably SSD for faster performance)

2.3. Time Line chart

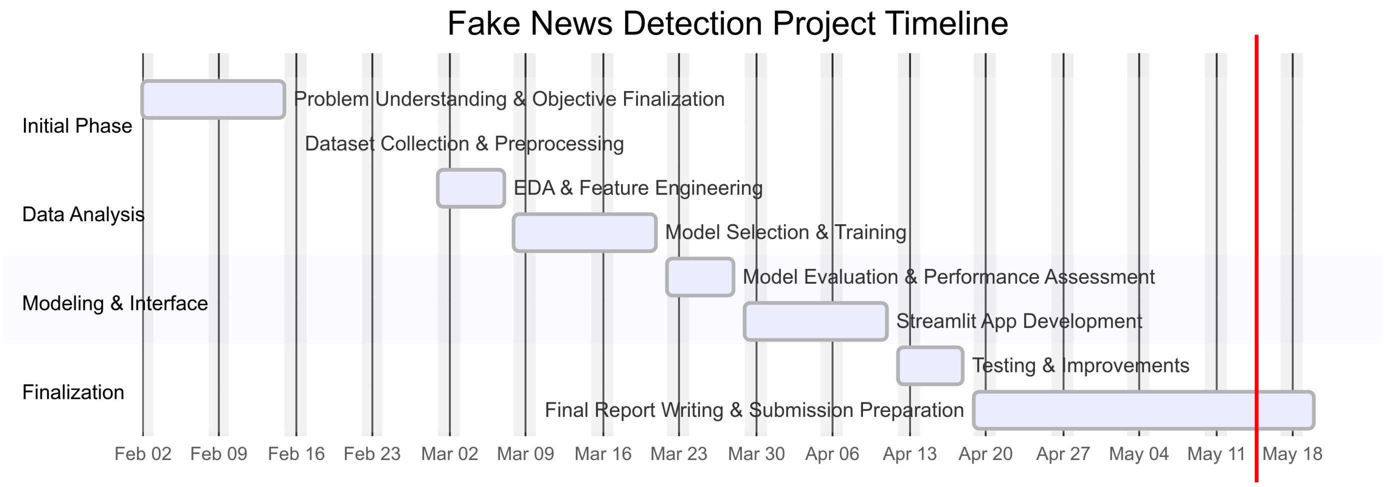

Figure 2.1: Project Timeline Chart

3. PROPOSED METHODOLOGY

3.1. System Architecture

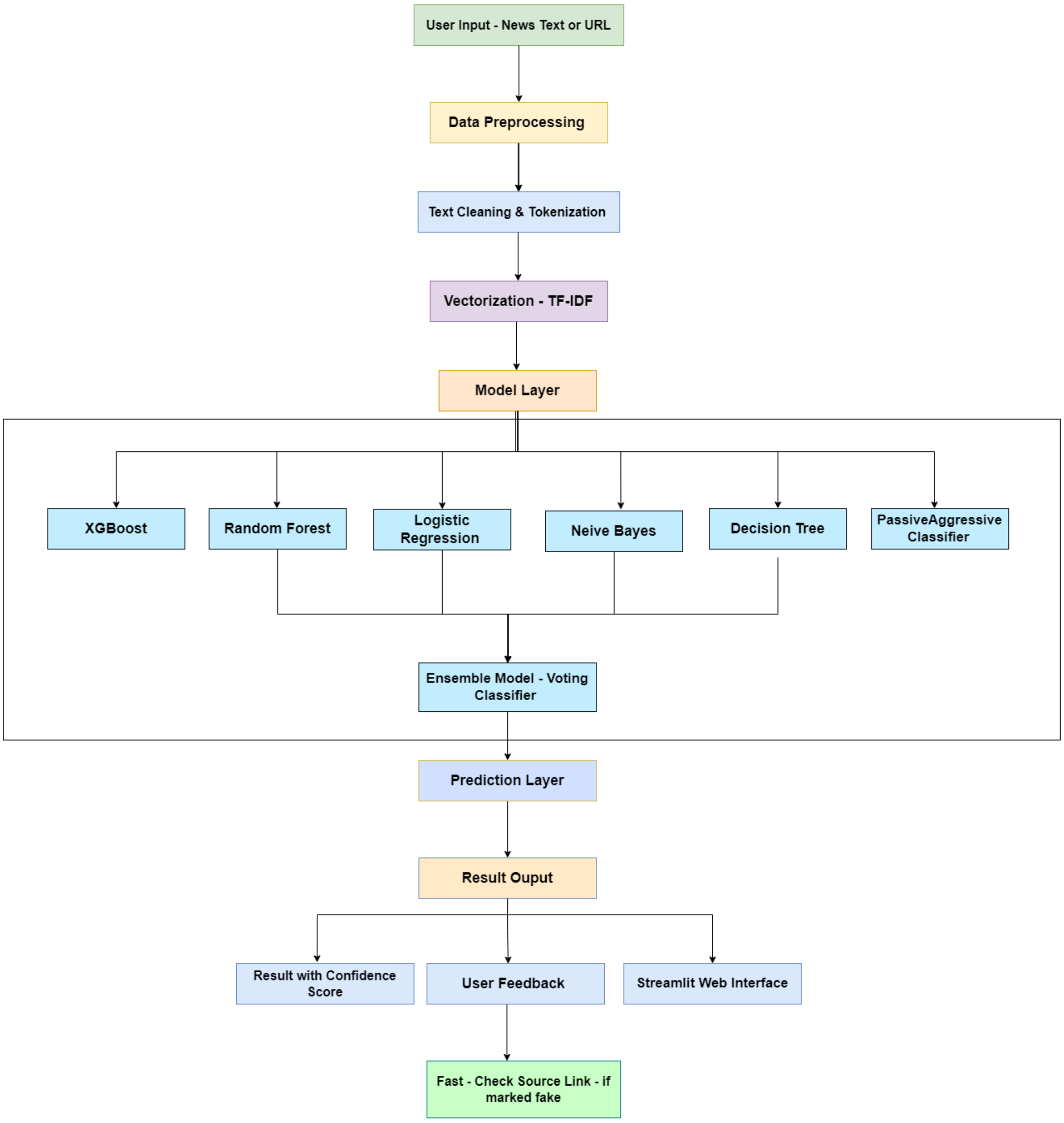
User Input - News Text or URL
Data Preprocessing
Text Cleaning & Tokenization
Vectorization - TF-IDF
Model Layer
XGBoost
Random Forest
Logistic Regression
Neive Bayes
Decision Tree
PassiveAggressive Classifier
Ensemble Model - Voting Classifier
Prediction Layer
Result Ouput
Result with Confidence Score
User Feedback
Streamlit Web Interface
Fast - Check Source Link - if marked fake

Figure 3.1: System Architecture

3.2. Methodology (Algorithms used)

This project implements a structured machine learning pipeline to detect fake news articles using textual input. The methodology involves key steps such as data preprocessing, feature extraction, model training, evaluation, and ensemble learning, ultimately deployed via a user-friendly Streamlit web interface.

- **Data Preprocessing:** The news data underwent multiple cleaning steps including removal of HTML tags, punctuation, stopwords, and lowercasing. This was followed by tokenization and lemmatization to standardize the text and reduce word complexity.

- **Feature Engineering:** The processed text data was converted into numerical format using the **TF-IDF (Term Frequency-Inverse Document Frequency)** vectorizer. This method assigns weights to words based on their importance in distinguishing real and fake news.

- **Model Building:** Several supervised machine learning models were developed and compared to identify the most accurate classifier for fake news detection. These include:

  - **Logistic Regression**
  - **Random Forest Classifier**
  - **Naive Bayes**
  - **Decision Tree Classifier**
  - **XGBoost Classifier**
  - **Gradient Boosting Classifier**
  - **Passive Aggressive Classifier**

- **Model Evaluation:** The performance of each model was evaluated using classification metrics such as:

  - Accuracy
  - Precision
  - Recall
  - F1-Score

  These metrics helped in identifying models with the best trade-off between false positives and false negatives, especially for detecting fake news.

- **Ensemble Learning:** To improve robustness and overall accuracy, an ensemble model was developed using a **Voting Classifier** that combined the predictions of the top four performing models:

  - Logistic Regression
  - Random Forest

  - Naive Bayes
  - Decision Tree

  Majority voting (hard voting) was used to derive the final prediction based on model consensus.

- **Model Explanation and Feedback:** To enhance interpretability, **LIME (Local Interpretable Model-Agnostic Explanations)** was integrated into the interface. It helps users understand which words influenced the model's prediction. Additionally, a user feedback section allows manual correction of the result, which can guide future improvements.

- **Deployment:** The final ensemble model, explanation mechanism, and feedback system were deployed using a **Streamlit web application**, enabling end users to interact with the system via manual text input or a news article URL.

This methodology ensures a comprehensive and interpretable solution for fake news detection, helping users identify misinformation effectively.

3.3. Pseduo code

The following pseudo code outlines the major steps in the fake news detection pipeline:

```
1. Load dataset (CSV file containing labeled news articles)

2. Data Preprocessing:
   - Remove unnecessary columns (e.g., IDs, titles, etc. if not relevant)
   - Handle missing values (e.g., fill blanks with appropriate defaults or drop)
   - Clean the news text:
       a. Remove punctuation, special characters, and HTML tags
       b. Convert text to lowercase
       c. Remove stopwords
       d. Apply lemmatization

3. Text Vectorization:
   - Convert cleaned text to numerical format using TF-IDF vectorizer

4. Split the dataset:
   - Divide data into training and test sets (e.g., 80% train, 20% test)
```

```
5. Initialize individual machine learning models:
   - Logistic Regression
   - Random Forest Classifier
   - Naive Bayes
   - Decision Tree Classifier
   - XGBoost Classifier
   - Passive Aggressive Classifier
   - Gradient Boosting Classifier

6. Train each model on the training data

7. Evaluate each model:
   - Calculate Accuracy, Precision, Recall, and F1-Score on the test data

8. Create Ensemble Model using Voting Classifier:
   - Combine Logistic Regression, Random Forest, Naive Bayes, and
     Decision Tree using Majority Voting (Hard Voting)

9. Evaluate the Ensemble Model:
   - Calculate final classification metrics on test data

10. Deploy model using Streamlit:
    - Accept user input: raw news text or article URL
    - Predict whether the news is REAL or FAKE
    - Display prediction with confidence score

11. Explain the model decision using LIME:
    - Show the important words influencing prediction

12. Collect user feedback:
    - Allow user to mark prediction as correct or incorrect
    - If incorrect, show fact-check source links
```

3.4. Design

The design of the Fake News Detection System adopts a modular and scalable architecture. It incorporates key stages of Natural Language Processing (NLP) and Machine Learning (ML), including data preprocessing, feature extraction, model training, evaluation, and result explanation. Each component in the system is designed to work independently while contributing to the overall prediction pipeline.

The system architecture supports both manual and real-time prediction workflows. Users can input individual news articles for classification through a user-friendly web interface, while internal components handle text cleaning, TF-IDF vectorization, and classification using trained machine learning models. The integration of LIME (Local Interpretable Model-Agnostic Explanations) further enhances transparency by visually explaining which features influenced the prediction.

This section includes data flow diagrams (DFD) and UML component diagrams that illustrate the architecture, data flow, and interaction between core modules within the fake news detection system.

3.4.1. Data Flow Diagrams

Level 0 DFD

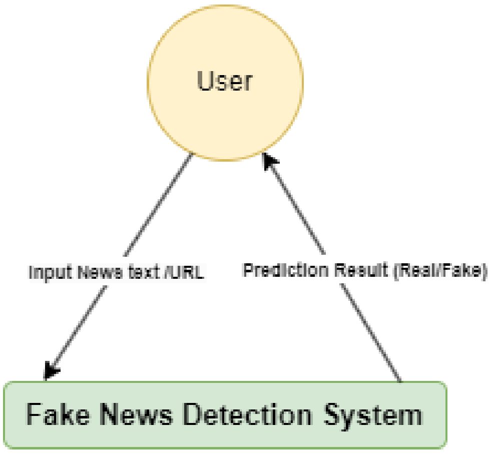

Fig. 3.2: Level 0 Data Flow Diagram for Fake News Detection

Level 1 DFD

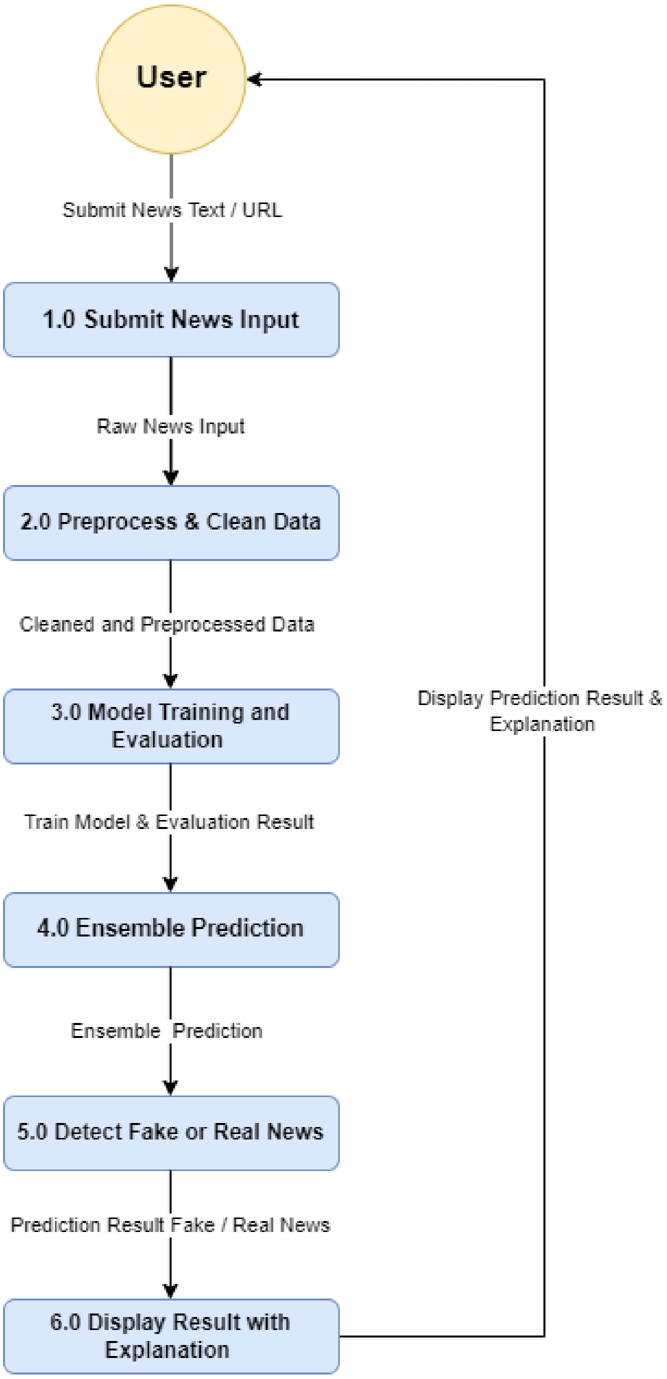
User
Submit News Text / URL
1.0 Submit News Input
Raw News Input
2.0 Preprocess & Clean Data
Cleaned and Preprocessed Data
3.0 Model Training and Evaluation
Train Model & Evaluation Result
4.0 Ensemble Prediction
Ensemble Prediction
5.0 Detect Fake or Real News
Prediction Result Fake / Real News
6.0 Display Result with Explanation
Display Prediction Result & Explanation

Fig. 3.3: Level 1 Data Flow Diagram showing internal components of the Fake news detection system

3.4.2. UML Diagrams

1. **Flow Chart Diagram**

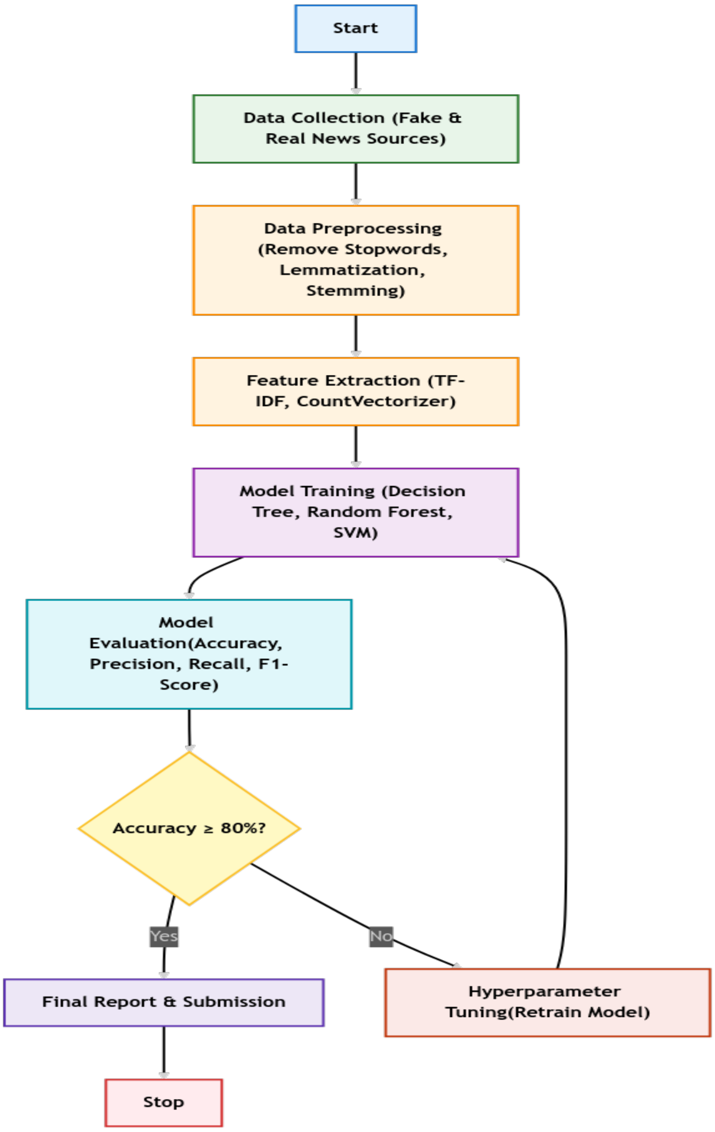
Start
Data Collection (Fake & Real News Sources)
Data Preprocessing (Remove Stopwords, Lemmatization, Stemming)
Feature Extraction (TF-IDF, CountVectorizer)
Model Training (Decision Tree, Random Forest, SVM)
Model Evaluation(Accuracy, Precision, Recall, F1-Score)
Accuracy ≥ 80%?
Yes
No
Final Report & Submission
Hyperparameter Tuning(Retrain Model)
Stop

Figure 3.4: Flowchart

2. **Use Case Diagram**

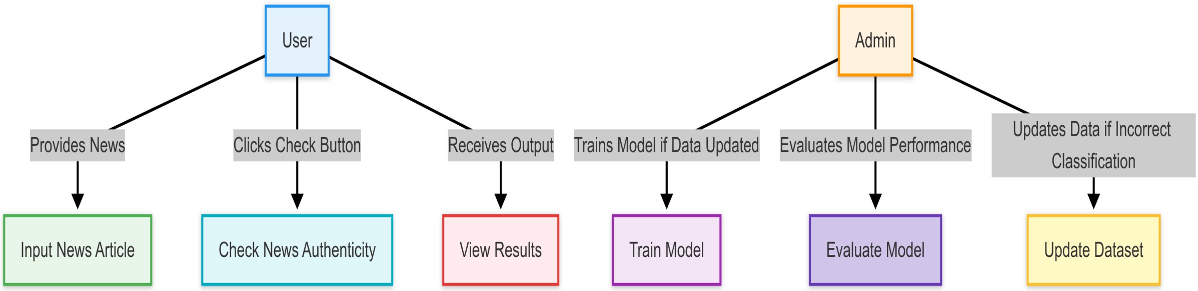

Figure 3.5: Use Case

3. Sequence Diagram

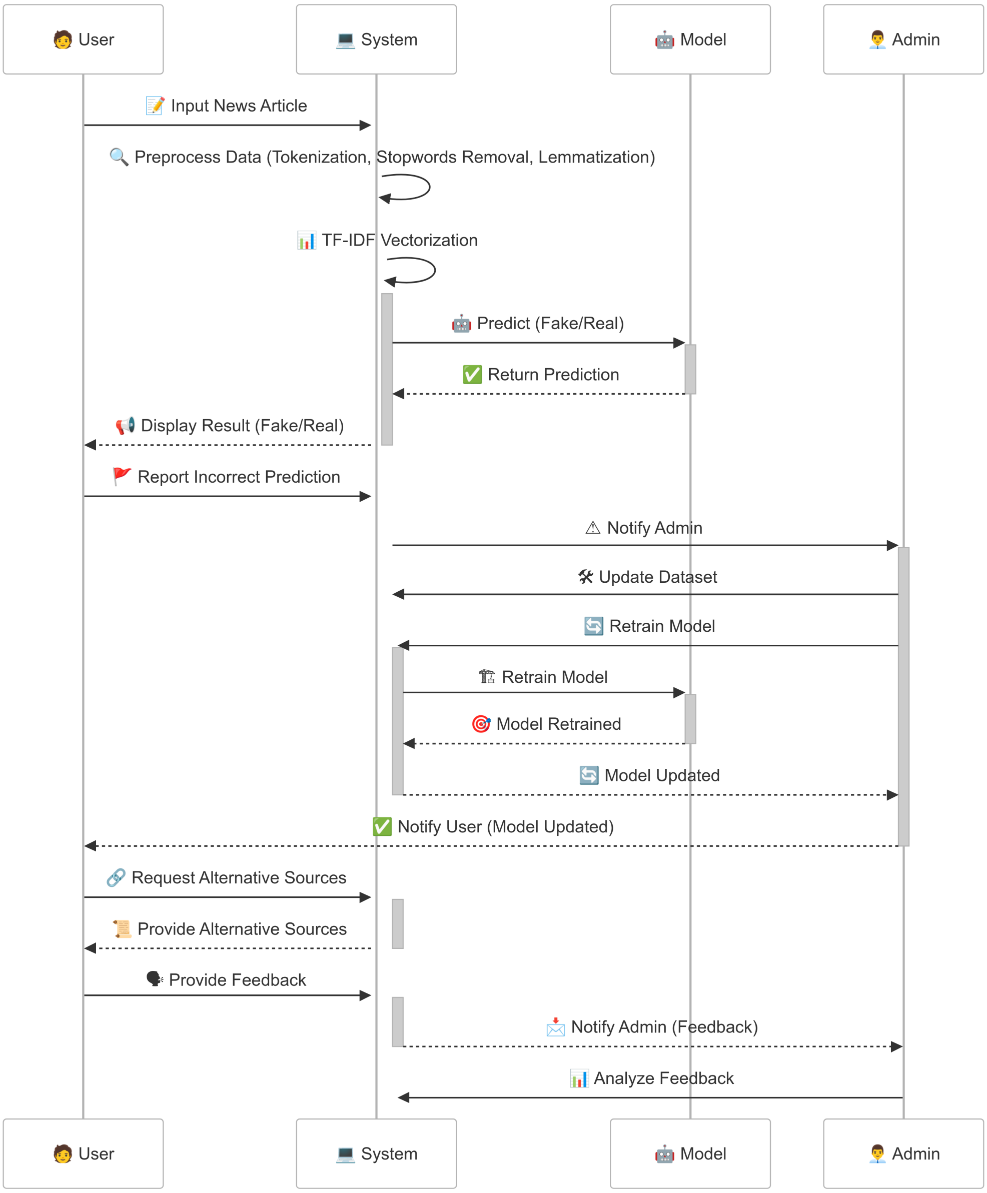

Figure 3.6: Sequence Diagram

4. **Acitvity Diagram**

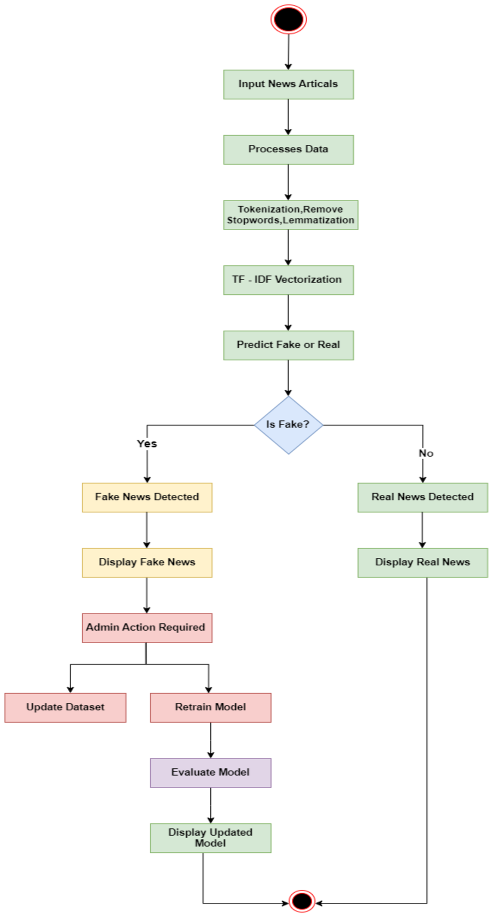

Figure 3.7: Activity Diagram

5. **Class Diagram**

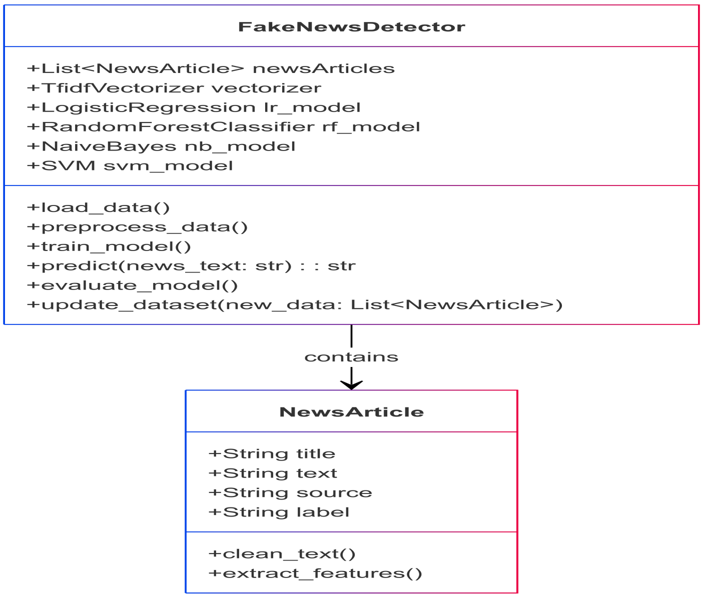

Figure 3.8: Class Diagram

4. RESULTS AND EXPLANATION

4.1. Implementation Approaches

The fake news detection system was implemented using a modular, interpretable, and scalable architecture. The entire pipeline is developed using Python and consists of multiple components such as data preprocessing, feature extraction, model training, evaluation, explanation, and deployment. Key libraries and tools used include:

- **Pandas** and **NumPy** for data manipulation and handling.
- **NLTK**, **TextBlob**, and **TextStat** for Natural Language Processing tasks such as lemmatization, sentiment scoring, and readability analysis.
- **Scikit-learn** for TF-IDF vectorization, classification models, and evaluation metrics.
- **Matplotlib** and **Seaborn** for Exploratory Data Analysis (EDA) and visualizations.
- **LIME** for interpreting model predictions using local explanations.
- **WordCloud** for visualizing the most frequent terms in fake and real news.
- **Streamlit** for deploying the fake news detection interface with user input support and real-time prediction results.

The implementation was carried out in the following stages:

1. **Data Collection:** Publicly available datasets of real and fake news articles were used.
2. **Data Preprocessing:** Removed noise, special characters, stopwords, and performed lemmatization to clean text data.
3. **Feature Extraction:** Applied TF-IDF for vectorizing the news text. Additional features like word count, character count, sentiment score, readability score, POS tagging, and NER were also extracted.
4. **Exploratory Data Analysis:** Analyzed data distributions, subject categories, text lengths, and generated WordClouds to observe word patterns.
5. **Model Training:** Implemented and trained six ML models: PassiveAggressiveClassifier, Naïve Bayes, Logistic Regression, Decision Tree, Random Forest, and an Ensemble Voting Classifier.

6. **Evaluation:** Compared models using Accuracy, Precision, Recall, and F1-score. The best models achieved over 95% accuracy.
7. **Explainability:** Integrated LIME (Local Interpretable Model-Agnostic Explanations) to explain individual predictions and identify influential words.
8. **Manual Testing:** Reserved sample records from both classes to validate predictions manually.
9. **Deployment:** Deployed the fake news detection system using **Streamlit**, allowing users to input news content and receive instant classification results through a simple web interface.

This modular and interpretable approach allows each component to be developed, evaluated, and deployed independently, resulting in a robust fake news detection pipeline ready for real-world applications.

4.2. Testing

Testing was conducted to assess the performance, reliability, and generalization capability of the developed fake news detection models. The dataset was split into an 80:20 ratio, with 80% of the data used for training and 20% reserved for testing.

The following testing strategies were applied:

- **Hold-out Validation:** The reserved 20% of the dataset was used to evaluate model performance on unseen news articles.
- **Performance Metrics:** Accuracy, Precision, Recall, and F1-score were used to evaluate the effectiveness of each classification model.
- **Confusion Matrix Analysis:** Confusion matrices were generated to observe classification correctness in terms of true/false positives and negatives.
- **Manual Testing:** A subset of news articles (10 fake and 10 real) was excluded during training and used to manually validate prediction results.

Among all trained models, the Ensemble Voting Classifier demonstrated the most consistent and accurate performance, achieving over 95% accuracy by combining the strengths of multiple base models.

Additionally, real-time testing was performed using the deployed **Streamlit** web application. Users could input a news article (or paste text) into the interface to receive instant predictions. The system returned the following:

- **Prediction Result:** Whether the news is classified as Real or Fake.
- **Prediction Probability:** Confidence score associated with the prediction.
- **LIME Explanation:** Highlighted influential words that contributed to the model's decision.

This comprehensive testing ensured that the fake news detection system was not only accurate and efficient but also transparent, interpretable, and ready for real-time deployment.

4.3. Analysis (graphs/chart)

The following section presents various visualizations and insights extracted from the dataset and model outputs to better understand the patterns, structure, and linguistic features of real and fake news articles in the fake news detection system.

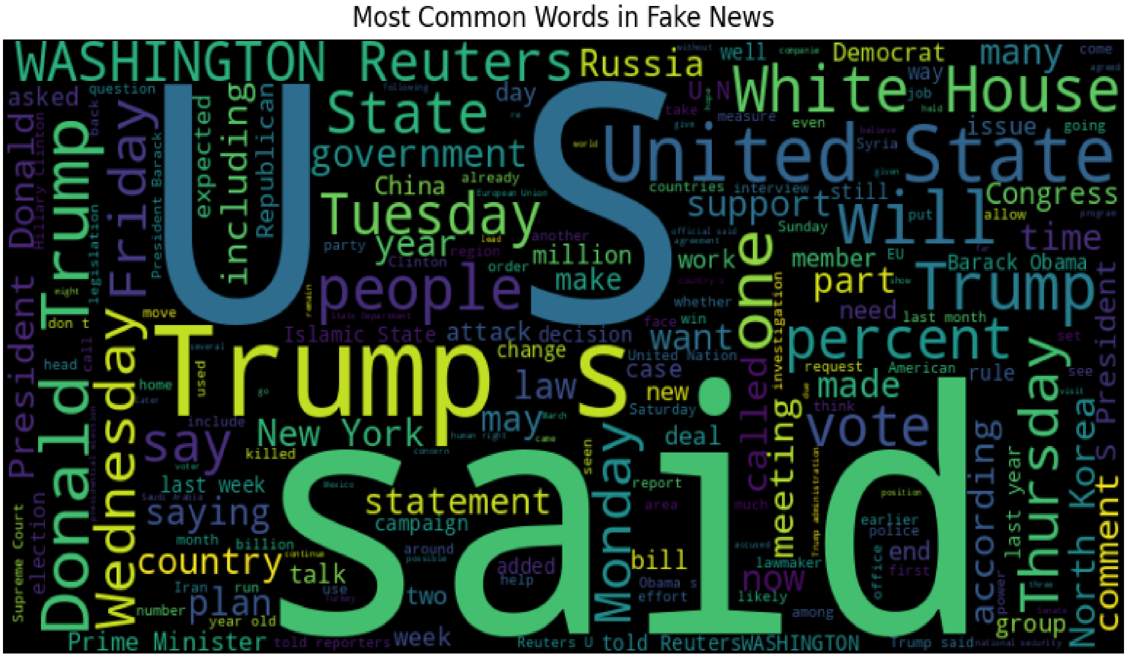

Figure 4.1: WordCloud – Common words in fake news

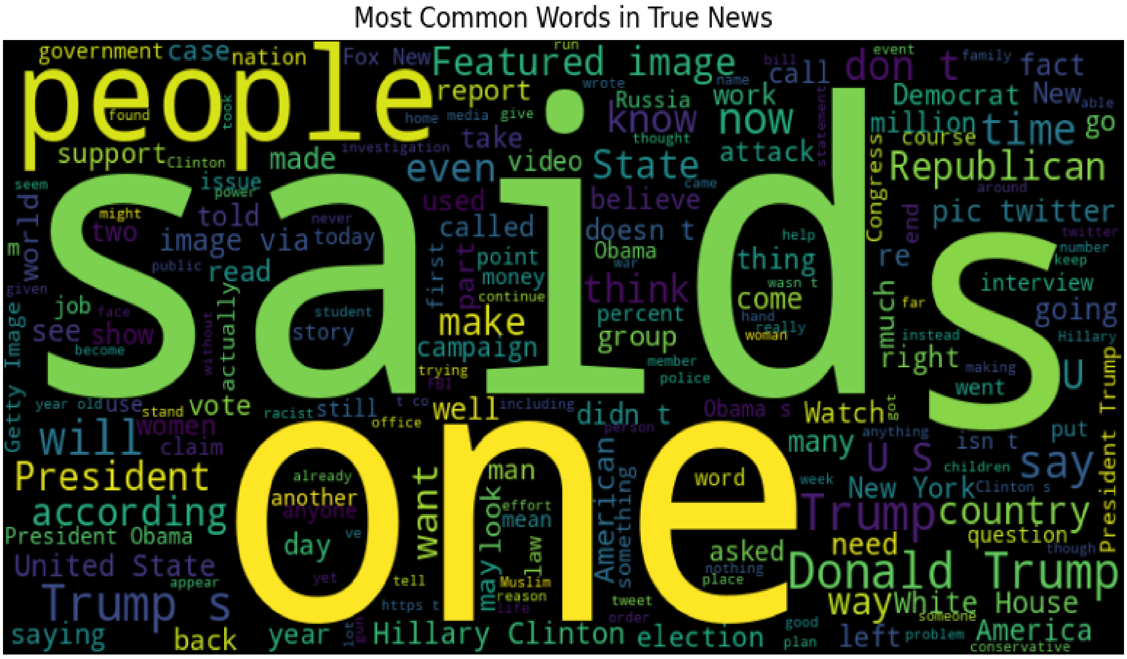

Figure 4.2: WordCloud – Common words in real news

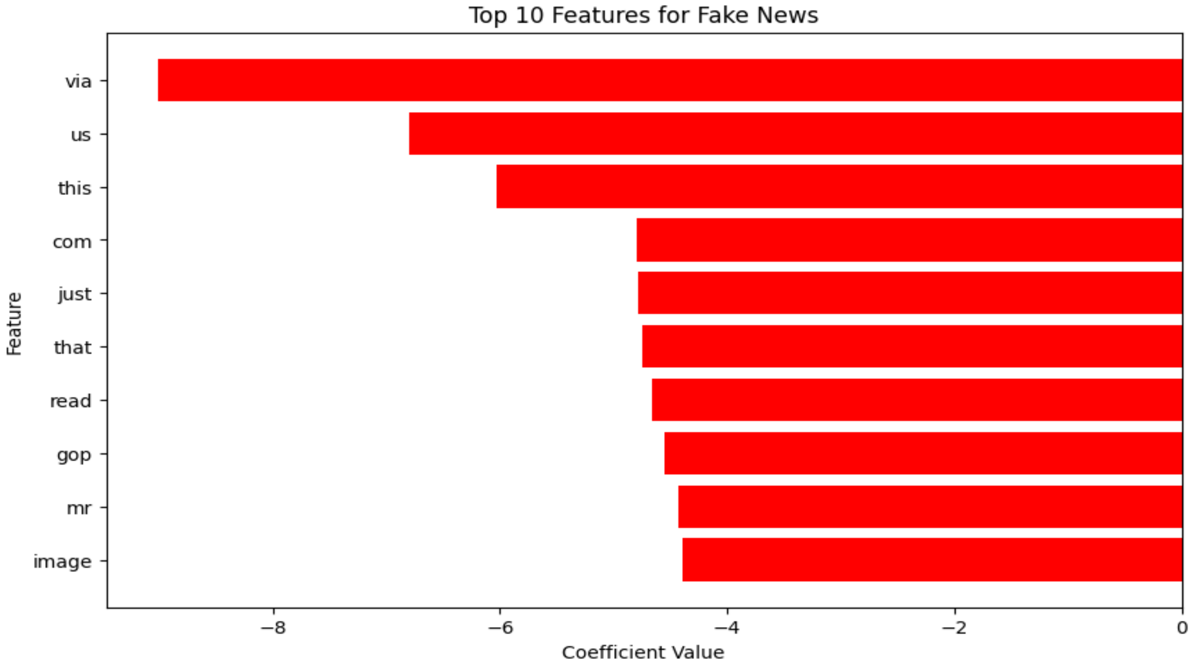

Figure 4.3: Top 10 features for fake news

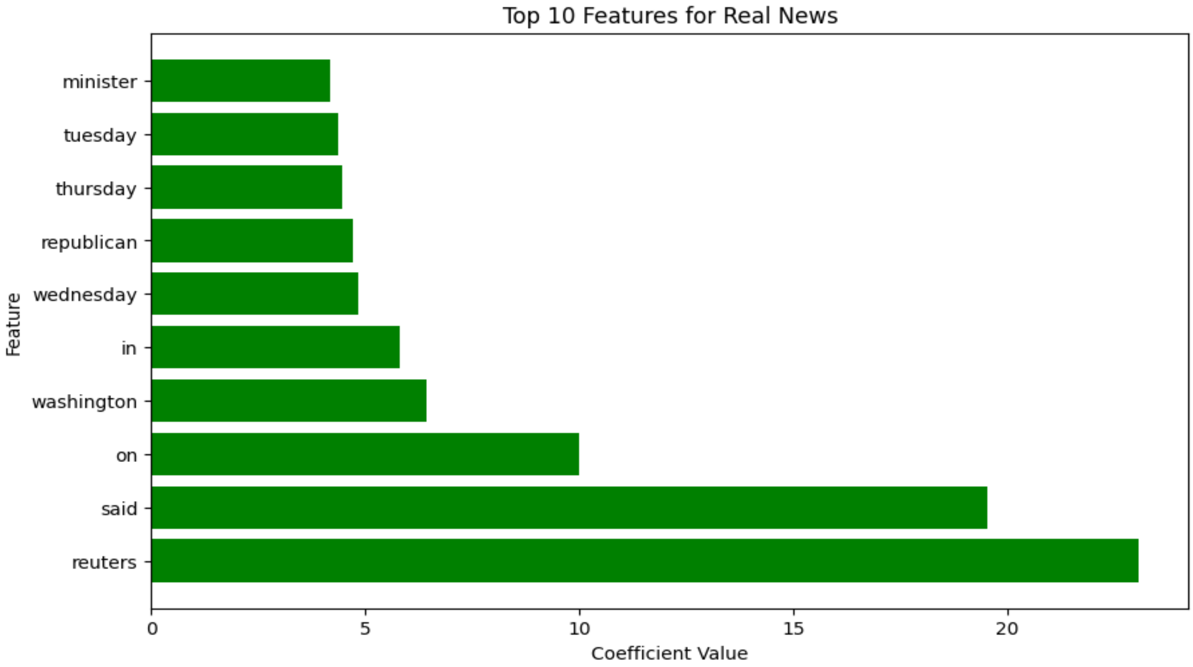

Figure 4.4: Top 10 features for real news

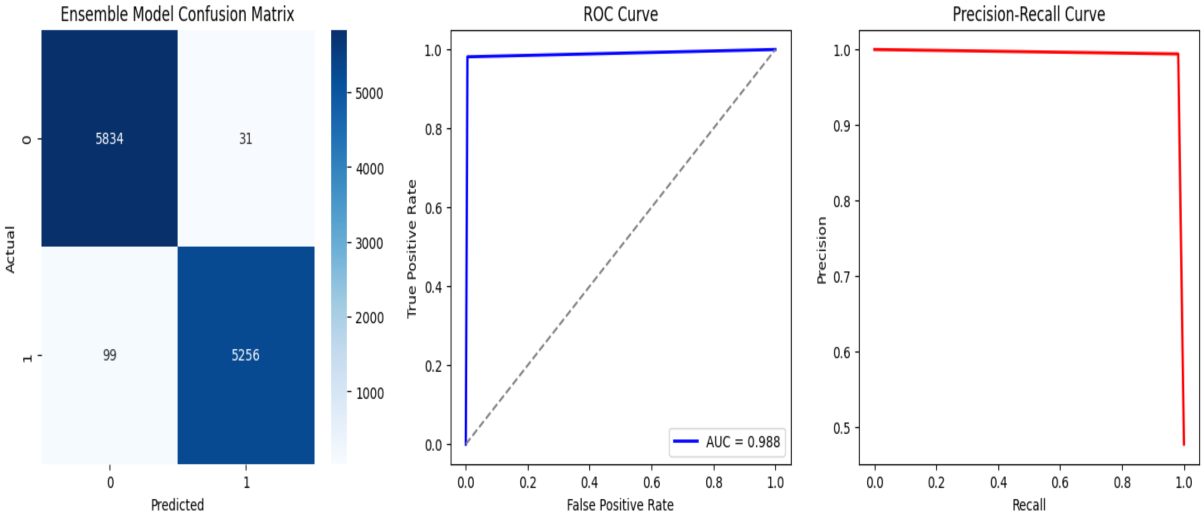

Figure X: Confusion Matrix, Precision/Recall, and ROC Curve for Ensemble Model

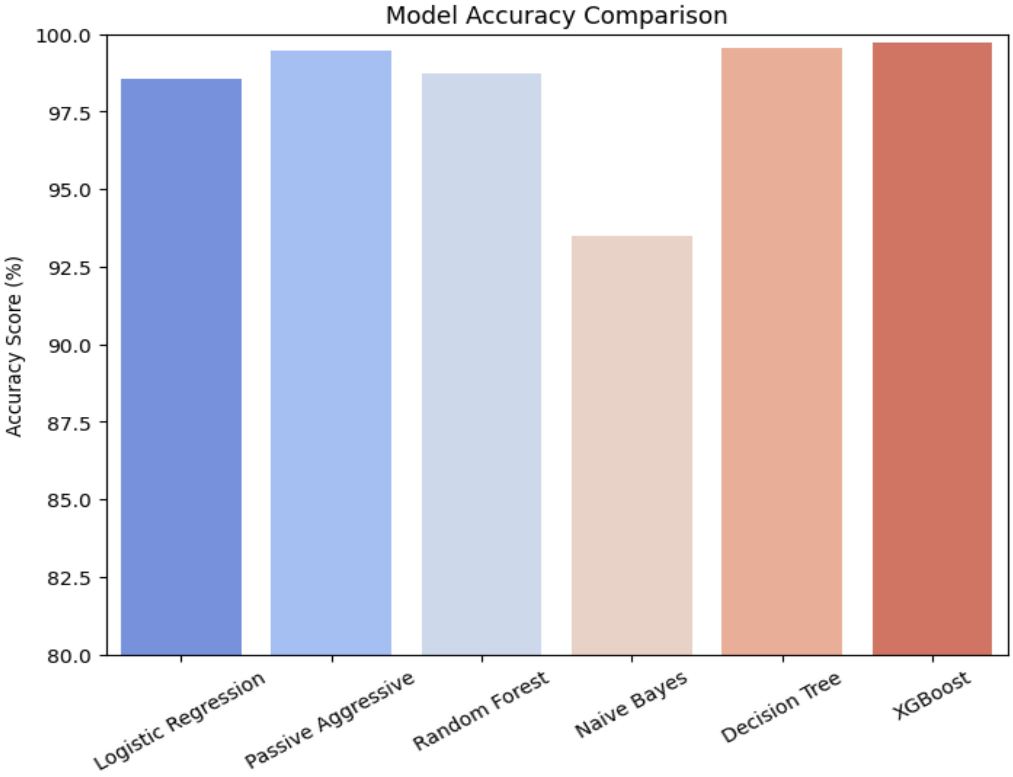

Figure X: Accuracy comparison of different machine learning models used in fake news detection

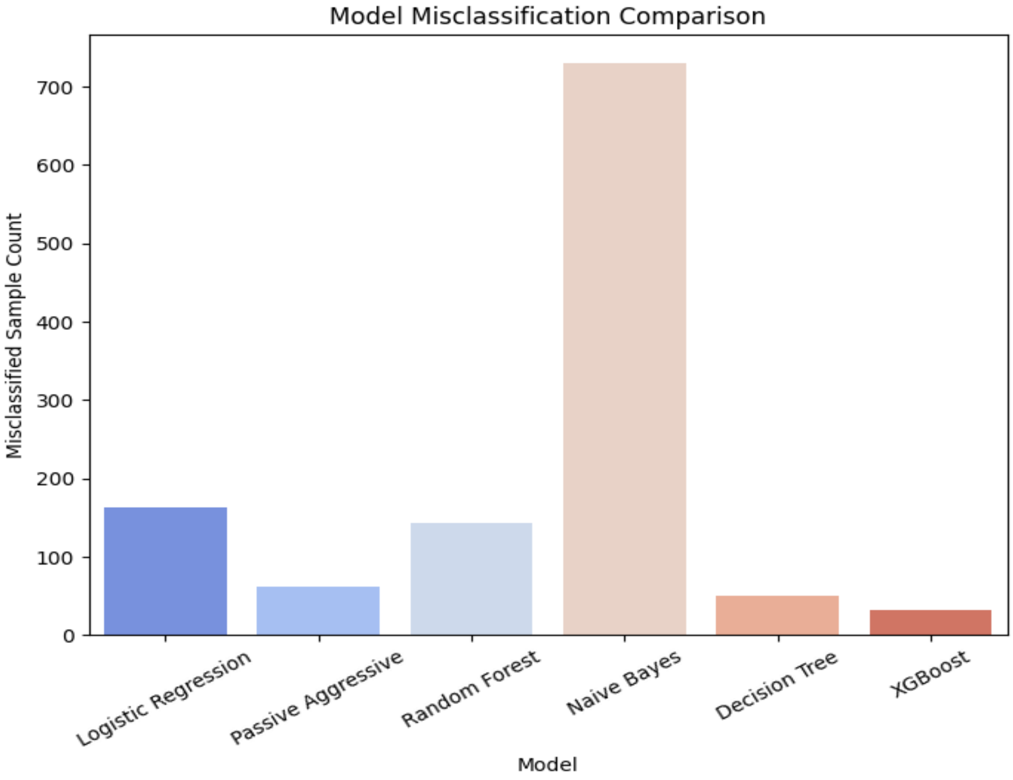

Figure X: Comparison of misclassifications across different machine learning models

5. CONCLUSION & FUTURE SCOPE

Conclusion

This project successfully demonstrated the use of various machine learning and natural language processing techniques to detect fake news articles. A comprehensive dataset was used to analyze language patterns and content structure. The data underwent preprocessing steps including text cleaning, stopword removal, lemmatization, and feature extraction using TF-IDF and linguistic features to enhance model performance.

Multiple machine learning models such as PassiveAggressiveClassifier, Naïve Bayes, Logistic Regression, Decision Tree, Random Forest, and an Ensemble Voting Classifier were implemented and evaluated. Among them, the ensemble model delivered the highest accuracy and robust performance across key metrics such as Accuracy, Precision, Recall, and F1-score. The system was deployed using Streamlit, enabling real-time fake news detection with an interactive user interface.

Through this approach, users can verify the credibility of news content, reduce misinformation, and make informed decisions about the information they consume.

Future Scope

Although the current system provides a strong baseline for fake news detection, there are several areas for future improvements:

- **Deep Learning Models:** Incorporating advanced models like transformers to better understand context and semantics.
- **Multilingual Support:** Extending detection capabilities to multiple languages for wider applicability.
- **Multimodal Analysis:** Including images, videos, and hyperlinks for more comprehensive detection.
- **Real-Time Monitoring:** Deploying real-time analysis on social media and news feeds to flag fake content instantly.

By incorporating these improvements, the system can be enhanced to serve as a comprehensive fake news detection solution tailored to dynamic and evolving information environments.

References

[1] I. Ahmad, M. Yousaf, S. Yousaf, and M. O. Ahmad, "Fake news detection using machine learning ensemble methods," *Complexity*, vol. 2020, no. 1, p. 8885861, 2020.

[2] Z. Khanam, B. Alwasel, H. Sirafi, and M. Rashid, "Fake news detection using machine learning approaches," in *IOP conference series: materials science and engineering*, vol. 1099, no. 1. IOP Publishing, 2021, p. 012040.

[3] S. Kumar, R. Asthana, S. Upadhyay, N. Upreti, and M. Akbar, "Fake news detection using deep learning models: A novel approach," *Transactions on Emerging Telecommunications Technologies*, vol. 31, no. 2, p. e3767, 2020.

[4] M. Aldwairi and A. Alwahedi, "Detecting fake news in social media networks," *Procedia Computer Science*, vol. 141, pp. 215–222, 2018.

[5] X. Zhou and R. Zafarani, "Fake news: A survey of research, detection methods, and opportunities," *arXiv preprint arXiv:1812.00315*, vol. 2, 2018.

[6] C. K. Hiramath and G. Deshpande, "Fake news detection using deep learning techniques," in *2019 1st International Conference on Advances in Information Technology (ICAIT).* IEEE, 2019, pp. 411–415.

[7] S. H. Kong, L. M. Tan, K. H. Gan, and N. H. Samsudin, "Fake news detection using deep learning," in *2020 IEEE 10th symposium on computer applications & industrial electronics (ISCAIE).* IEEE, 2020, pp. 102–107.

[8] L. Waikhom and R. S. Goswami, "Fake news detection using machine learning," in *Proceedings of international conference on advancements in computing & management (ICACM)*, 2019.

[9] A. A. Tanvir, E. M. Mahir, S. Akhter, and M. R. Huq, "Detecting fake news using machine learning and deep learning algorithms," in *2019 7th international conference on smart computing & communications (ICSCC).* IEEE, 2019, pp. 1–5.

[10] A. Abdulrahman and M. Baykara, "Fake news detection using machine learning and deep learning algorithms," in *2020 international conference on advanced science and engineering (ICOASE).* IEEE, 2020, pp. 18–23.