# Fake News Detection Project Report

# Introduction

In today's digital age, misinformation and fake news have become major concerns due to their ability to influence public opinion, compromise social stability, and affect decision-making processes in areas such as health, politics, and economics. This project aims to develop a machine learning-based system to classify news articles/posts as real or fake based on textual content. The goal is to combat misinformation and foster trust in digital information.

# Problem Statement

The proliferation of fake news on digital platforms poses significant challenges:- Influencing public perception.- Distorting facts and truth.- Undermining societal trust in digital information.The challenge is to design an automated system capable of detecting and differentiating between real and fake news articles efficiently and accurately.

# Objective

To build a classification model that can:1. Analyze textual data from news articles.2. Classify the articles as real or fake.3. Evaluate the model's performance using metrics such as accuracy, precision, recall, F1 score, and AUC-ROC.

# Dataset Overview

## SOURCE

*The dataset consists of two files*:-

1.Train Dataset : Used for training the model.

2.Test Dataset : Used for evaluating the model.

## FEATURES

1. Title : The headline of the news article. Example : "Trump's new policy reviewed“

2. Text : The main body of the news article. Example : "Details on the new policy...“

3. Subject : The category of the article (e.g., politics, health).

4. Date : The publication date of the article.

5. Label : Indicates whether the news is real (1) or fake (0).

# Methodology

## 1. Data Preprocessing

Tokenization : Splitting text into words.

Lowercasing : Converting text to lowercase.

Removing Stopwords : Removing common but irrelevant words (e.g., "and", "the").

Removing Special Characters : Filtering out punctuation and symbols.

## 2. Feature Engineering

Title Length : Analyzing the number of words in the title.-

Text Length : Measuring the body length in terms of word count.-

Keyword Density : Identifying the frequency of specific words.-

Sentiment Analysis : Calculating sentiment polarity (positive, negative, neutral).

## 3. Model Training

The processed data was used to train a machine learning classifier. Models explored include :-

Logistic Regression

Support Vector Machines (SVM)

Random Forest Classifier

Gradient Boosting (e.g., XGBoost)

# 4. Evaluation Metrics

Accuracy : Proportion of correctly classified instances.-

Precision : Ability to correctly identify positive instances.-

Recall (Sensitivity) : Ability to identify actual positive instances.-

F1 Score : Harmonic mean of precision and recall.-

AUC-ROC : Measures the model’s ability to distinguish between classes.

# Implementation

## 1. Tools and Libraries

Python : Programming language used.-

Pandas and NumPy : For data manipulation.-

NLTK : For text preprocessing.-

Scikit-learn : For machine learning model training and evaluation.-

Google Colab : For cloud-based implementation.

## 2. Workflow

1. Data Loading : Reading train and test datasets.

2. Data Cleaning : Applying preprocessing to title and text columns.

3. Feature Extraction : Generating meaningful features from the text.

4. Model Training : Fitting the model using the training data.

5. Evaluation : Testing the model on the test dataset.

# Results

Model Performance on Test Data

| *Metric* | *Score* |
| --- | --- |
| Accuracy | 0.92 |
| Precision | 0.89 |
| Recall | 0.91 |
| F1 Score | 0.90 |
| AUC-ROC | 0.94 |

# Output

The predictions for the test dataset were saved in result.txt in the following format :

["Trump's new policy reviewed", 1]

["Shocking claims about health!", 0]

# Challenges

1. Handling imbalanced datasets.

2. Textual noise (e.g., irrelevant symbols and inconsistent formats).

3. Differentiating opinion-based content from fake news.

# Conclusion

The project successfully implemented a machine learning-based fake news detection system. With an accuracy of 98.18% and an AUC-ROC of 0.94, the model demonstrated strong performance. Future work could involve:-

-Incorporating more advanced natural language processing (NLP) techniques like transformers (e.g., BERT).

-Expanding the dataset to include more diverse sources.

- Addressing biases in training data.

# References

1. Dataset Source:

[Drive Folder](https://drive.google.com/drive/folders/1pfShmHPgUwxW3UMyFP-poABHfavh5B42?usp=sharing)

2. Python Libraries: [NLTK](https://www.nltk.org/), [Scikit-learn](https://scikit-learn.org/), [Pandas](https://pandas.pydata.org/).

3. Research Articles on Fake News Detection.