FAKE NEWS DETECTION USING TF-IDF AND PASSIVE AGGRESSIVE CLASSIFIER

Submitted by

D.MANVITHA(RA2311027020115)
A.GNANASREE(RA2311027020076)
CH.NAGA PUJITHA(RA2311027020102)

Under the guidance of

Dr.Revathy.S

Assistant Professor

Department of Computer Science and Engineering

In partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

SPECIALISATION WITH BIG DATA ANALYTICS

of

FACULTY OF ENGINEERING AND TECHNOLOGY

CHENNAI RAMAPURAM
SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

RAMAPURAM, CHENNAI-600089

APRIL 2025

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Deemed to be University Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that the Mini Project titled "**Fake News Detection using TF-IDF and Passive Aggressive Classifier**" is the bonafide certificate of **D.MANVITHA - RA2311027020115, A.GNANASREE - RA2311027020076, CH.NAGA PUJITHA-RA2311027020102** of II Year CSE-BDA submitted for the course 21CSC206T – Artificial Intelligence for the Academic Year 2024 – 25 Even Semester.

SIGNATURE

Dr.Revathy.S ,M.E,PhD.,

Assistant Professor

Department of Cyber Security

School of Computer Science and Engineering

SRM Institute of Science and Technology

Ramapuram

Chennai 89.

TABLE OF CONTENTS

ABSTRACT

The exponential growth of digital media has led to an alarming rise in the spread of fake news, which can influence public opinion, manipulate elections, and cause social unrest. This project presents a comprehensive fake news detection system that leverages Natural Language Processing (NLP) and machine learning to automatically identify and classify news articles based on their authenticity. The approach involves collecting and cleaning large volumes of news data from reliable datasets, followed by preprocessing steps such as tokenization, stop-word removal, and stemming. Feature extraction techniques like TF-IDF and word embeddings are used to convert text into meaningful vectors. These vectors are then fed into various classification algorithms, including Logistic Regression, Random Forest, Support Vector Machine, and advanced deep learning models like LSTM and BERT, to detect patterns indicative of deceptive content. The system is rigorously trained and evaluated using performance metrics such as accuracy, precision, recall, and F1-score, achieving strong results that demonstrate its effectiveness. Ultimately, this project aims to offer a scalable and real-time solution to combat misinformation, enhance media literacy, and support the integrity of online information ecosystems.

INTRODUCTION

In the modern digital era, the rapid evolution of the internet and social media platforms has revolutionized the way information is produced, shared, and consumed. News, which was once disseminated primarily through trusted print and broadcast media, is now widely available on online platforms, often without stringent editorial oversight. While this shift has significantly improved the accessibility of information, it has also paved the way for a serious issue—the spread of fake news.

Fake news refers to misinformation or hoaxes that are deliberately spread to mislead people or gain political, financial, or social advantage. These fabricated stories may appear credible and are often designed to provoke emotional responses, polarize opinions, and misinform the public.

With the increasing reach of social networking platforms such as Facebook, Twitter, and WhatsApp, fake news can go viral within minutes, affecting millions of users before fact-checking mechanisms can respond. The consequences of such misinformation are far-reaching, including public panic, damaged reputations, political manipulation, and even violence in some cases.

The need for automated fake news detection systems has therefore become more pressing than ever. Human fact-checking, while effective, is time-consuming and not scalable to the vast amount of information being generated every day. To address this, researchers have turned to the field of Natural Language Processing (NLP) and Machine Learning (ML) to develop systems that can detect and classify fake news based on textual analysis.
These technologies can be used to build intelligent models capable of learning from labeled datasets and making accurate predictions about the authenticity of new or unseen content.

This project aims to develop a fake news detection system that automatically classifies news articles or social media posts as either real or fake. The project workflow involves multiple stages, starting with the collection of news data from trusted public datasets. This raw text data is then cleaned and preprocessed using standard NLP techniques such as tokenization, stop-word removal, lemmatization, and vectorization through methods like TF-IDF or word embeddings (e.g., Word2Vec, GloVe).

The processed data is used to train machine learning models including Logistic Regression, Decision Trees, Support Vector Machines (SVM), Random Forests, and more recently, deep learning models like Long Short-Term Memory (LSTM) networks and Bidirectional Encoder Representations from Transformers (BERT).

Each model is trained and tested on labeled datasets—such as the LIAR dataset or the FakeNewsNet corpus—to evaluate performance. Metrics such as accuracy, precision, recall, and F1-score are used to measure how well the system can differentiate between real and fake news. Deep learning models like LSTM and BERT, in particular, show high accuracy due to their ability to capture semantic meaning and contextual relationships within text.

The significance of this work lies in its potential to provide a scalable and real-time solution to detect misinformation at its source, especially in high-traffic digital environments. By flagging suspicious content early, such a system could support social media platforms, news aggregators, and even governmental agencies in curbing the spread of false information.

Additionally, it can serve as an educational tool to help users become more aware of how to distinguish between credible and deceptive content online.

In conclusion, fake news is a complex and growing challenge that demands a combination of technological innovation and societal awareness. Through the integration of NLP and machine learning, this project offers a promising step toward automating the detection of fake news and protecting the integrity of digital information ecosystems.

As part of the broader effort to combat misinformation, the system developed in this project can serve as a foundation for future enhancements involving multimodal data (text, images, videos), multilingual detection, and real-time deployment at scale.

SCOPE AND MOTIVATION

Scope

The scope of this project is centered around the development of an intelligent system capable of identifying and classifying fake news based on its textual content. The project leverages Natural Language Processing (NLP) and Machine Learning (ML) techniques to analyze and extract patterns from news articles, headlines, or social media posts to distinguish between authentic and deceptive information. This system is primarily focused on English-language text data and relies on supervised learning models trained on labeled datasets to perform binary classification—labeling a piece of news as either "Real" or "Fake."

The system is designed to handle the following core tasks:

1. Data Collection and Preprocessing: Collecting news articles from publicly available datasets (e.g., LIAR, FakeNewsNet), and cleaning and preparing the data for further processing using NLP techniques such as tokenization, stop-word removal, stemming/lemmatization, and vectorization (TF-IDF, Word2Vec, etc.).
2. Feature Extraction: Converting raw text into numerical representations that can be processed by ML algorithms. This includes extracting linguistic, lexical, and contextual features that may indicate deception.
3. Model Training and Evaluation: Applying machine learning classifiers such as Logistic Regression, Decision Trees, Random Forests, Naive Bayes, Support Vector Machines, and deep learning models like LSTM and BERT to train the fake news detection model. The models are evaluated using metrics such as accuracy, precision, recall, and F1-score to determine their effectiveness.
4. Prediction and Real-time Application: Once trained, the system should be capable of taking new, unseen text as input and predicting whether it is likely to be fake or real. The ultimate goal is to implement a scalable model that can be integrated into web platforms, social media applications, or browser extensions for real-time news verification.

While the current implementation is text-based and limited to binary classification, the scope can be extended in future iterations to include:

- Multilingual fake news detection.
- Multimodal detection (analyzing images, videos, or URLs along with text).
- Detection of satire and clickbait.

Motivation

The motivation behind this project stems from the increasingly negative impact of fake news on individuals, societies, and even nations. In today's hyperconnected world, people rely heavily on the internet and social media platforms to stay informed. While this provides unprecedented access to information, it also makes users highly vulnerable to misinformation. Fake news can have severe consequences, including spreading panic during crises, influencing elections, damaging reputations, and inciting violence or hate.

A few real-world incidents highlight the urgent need for automated fake news detection:

- During the COVID-19 pandemic, misinformation about cures, vaccines, and government policies circulated widely, leading to public confusion and health risks.
- Political fake news has been used to manipulate voter opinions, disrupt democratic processes, and create societal division.
- False news articles have contributed to financial fraud and hoaxes, affecting markets and personal investments.

Manual fact-checking by journalists and organizations, while effective, is slow and cannot keep up with the scale at which information is generated and shared. This creates a demand for automated, AI-powered solutions that can help detect and prevent the spread of misinformation at scale.

As a computer science student or AI practitioner, building such a system presents an opportunity to apply technical knowledge to a real-world problem with significant social relevance. The use of machine learning and NLP in this context is not just an academic exercise—it is a practical solution to a modern crisis. Developing a fake news detection system is a step toward empowering individuals with tools to make informed decisions, restoring trust in digital platforms, and enhancing information literacy in society.

Furthermore, this project offers rich learning outcomes. It covers diverse technical domains such as data collection and analysis, natural language understanding, supervised learning, model evaluation, and software development. It also opens doors for interdisciplinary collaboration with fields like journalism, psychology, and communication studies, where understanding the nature and impact of misinformation is critical.

PROJECT DESCRIPTION

The rise of digital platforms has transformed the way people consume news and information. While this has made access to knowledge easier and faster, it has also led to a growing concern—the proliferation of fake news. Fake news refers to misleading, fabricated, or intentionally false content, often published to manipulate public perception or generate sensationalism. As traditional journalism is replaced by algorithm-driven content delivery, misinformation can easily go viral and cause significant harm. The aim of this project is to develop an automated system that can detect and classify fake news using Natural Language Processing (NLP) and Machine Learning (ML) techniques.

2. System Overview

The Fake News Detection system consists of several interconnected modules, forming a complete pipeline from raw text input to classification output. The major components of the system include:

- Data Collection
- Data Preprocessing
- Feature Extraction
- Model Training
- Model Evaluation
- Prediction Interface

This modular design ensures flexibility, scalability, and ease of enhancement in future iterations.

3. Data Collection

The foundation of any machine learning model is quality data. For this project, publicly available datasets such as the LIAR dataset, FakeNewsNet, and Kaggle's Fake News Dataset are used. These datasets typically contain labeled samples of news headlines or articles with binary labels such as "real" or "fake".

Example fields in the dataset:

- Title: Headline of the news article.
- Text: Body content of the news.
- Label: Ground truth (real/fake).

The datasets are split into training, validation, and test sets in an 80:10:10 or 70:15:15 ratio for robust model training and evaluation.

4. Data Preprocessing

Preprocessing is crucial in NLP tasks to clean and normalize textual data before feeding it into machine learning algorithms. The following steps are applied:

1. Lowercasing: Converts all text to lowercase for uniformity.
2. Removal of Special Characters: Cleans punctuation, numbers, and symbols.
3. Tokenization: Breaks down the text into individual words or tokens.
4. Stop-word Removal: Removes common words like “the”, “is”, etc., that do not contribute to meaning.
5. Stemming or Lemmatization: Reduces words to their base or root form (e.g., “running” → “run”).
6. Vectorization: Transforms text into numerical format using:
    - Bag-of-Words (BoW)
    - TF-IDF (Term Frequency-Inverse Document Frequency)
    - Word Embeddings like Word2Vec or GloVe (for deep learning models)

5. Feature Extraction

Text cannot be processed directly by ML models, so feature extraction converts it into vector representations. Depending on the algorithm, different methods are used:

- TF-IDF: Assigns importance scores to words based on their frequency in a document vs. across all documents.
- Word2Vec/GloVe: Captures contextual meaning and semantic similarity by converting words into dense vectors.
- BERT Embeddings: Uses transformer-based models for contextualized word understanding, especially for deep learning.

6. Machine Learning Models

Multiple models are used and compared to identify the most effective approach:

6.1 Traditional ML Models

- Logistic Regression: Simple, interpretable baseline classifier.
- Naive Bayes: Efficient for large text datasets assuming independence between features.
- Support Vector Machine (SVM): Effective in high-dimensional spaces and for binary classification.
- Random Forest: An ensemble method using multiple decision trees.

6.2 Deep Learning Models

- LSTM (Long Short-Term Memory): Recurrent neural network model capable of learning long-term dependencies.
- BERT (Bidirectional Encoder Representations from Transformers): Pretrained language model from Google that captures rich contextual information.

Each model is trained on the dataset, and hyperparameters are tuned using cross-validation to optimize performance.

7. Evaluation Metrics

To evaluate model performance, the following metrics are used:

- Accuracy: Percentage of correctly classified samples.
- Precision: How many predicted fake news articles were actually fake.
- Recall: How many actual fake news articles were correctly identified.
- F1-Score: Harmonic mean of precision and recall.

Confusion matrices are also used to visualize the classification results and identify areas of improvement.

ALGORITHM OR PSEUDOCODE

The Fake News Detection system is developed using a combination of Natural Language Processing (NLP) techniques and Machine Learning (ML) models, both traditional and deep learning-based. This chapter provides a detailed theoretical understanding of how the system functions, beginning from the preprocessing of raw data to the final classification output. The logic is structured into multiple phases that together form an end-to-end automated detection pipeline.

1. Data Preprocessing Algorithm

The first and most critical phase in any text-based ML system is preprocessing. Raw textual data often contains inconsistencies such as capitalization, punctuation, noise, and irrelevant tokens that need to be cleaned before further analysis. The algorithm for preprocessing includes the following steps:

- Lowercasing: Converts all text to lowercase, ensuring uniformity across words and avoiding duplication of tokens (e.g., "Fake" and "fake" are treated the same).
- Special Character and Punctuation Removal: Non-alphanumeric characters like hashtags, symbols, numbers, and emojis are removed as they do not contribute meaningfully to the model.
- Tokenization: This step breaks down the text into individual components known as tokens, generally words. These tokens form the base elements for feature extraction.
- Stop Word Removal: Words like "the", "is", "and", which occur frequently but carry minimal semantic value, are removed from the token list.
- Lemmatization/Stemming: This transforms each token to its root form, which helps reduce vocabulary size while preserving meaning (e.g., "running" becomes "run").

This entire sequence is applied to each document in the dataset, transforming unstructured text into structured, normalized text that can be processed mathematically.

2. Feature Extraction Theory

Once preprocessing is complete, the cleaned text must be converted into a numerical format suitable for machine learning models. This transformation is achieved through feature extraction techniques.
The most widely used method in traditional models is Term Frequency–Inverse Document Frequency (TF-IDF). TF-IDF quantifies the importance of a word in a document relative to the entire corpus. Terms that appear frequently in one document but not across others are given higher importance. The

result is a sparse matrix where each row represents a document and each column represents a word, with the cell values indicating their TF-IDF scores.

In deep learning approaches, instead of TF-IDF, word embeddings such as Word2Vec, GloVe, or contextual embeddings like BERT are used. These embeddings map each word to a dense vector in a high-dimensional space, capturing not only the identity of the word but also its meaning and context in a sentence. This improves semantic understanding and enables the model to detect nuances in language.

3. Machine Learning Algorithms

Once features are extracted, the model must learn patterns to distinguish between fake and real news. Various machine learning classifiers can be employed for this task:

Logistic Regression
This is a baseline linear classifier that models the probability of an instance belonging to a class. In binary classification, it uses the sigmoid function to output a probability between 0 and 1, which is then thresholded to assign class labels. Despite its simplicity, logistic regression performs well in text classification when combined with TF-IDF vectors.

Support Vector Machine (SVM)
SVM is effective in high-dimensional spaces and is well-suited for binary classification. It works by identifying a hyperplane that best separates the classes, maximizing the margin between the closest points of the classes.

Naive Bayes
Particularly effective for text classification, this probabilistic model assumes independence between features and calculates the posterior probability of a class based on Bayes' theorem. It is simple, fast, and interpretable.

Random Forest
An ensemble method that builds multiple decision trees and outputs the mode of their predictions. It reduces overfitting and improves model robustness.

Each model is trained using a labeled training dataset and validated using a held-out test dataset. The results are evaluated based on metrics such as accuracy, precision, recall, and F1-score.

4. Deep Learning Models

For more advanced performance, deep learning models are used, particularly LSTM and BERT.

LSTM (Long Short-Term Memory)
LSTM is a type of Recurrent Neural Network (RNN) designed to remember long-term dependencies in sequence data. Unlike traditional RNNs, LSTMs are capable of overcoming the vanishing gradient problem through the use of gated memory units. In fake news detection, LSTMs can analyze the word order and sequence in a news article to capture context and hidden patterns.
A typical LSTM model for this task includes an embedding layer to convert words into dense vectors, an LSTM layer with memory cells, and a final dense layer with a sigmoid activation to output a binary prediction.

BERT (Bidirectional Encoder Representations from Transformers)
BERT is a state-of-the-art transformer model developed by Google. Unlike unidirectional models, BERT considers context from both directions (left and right) in a sentence, making it exceptionally good at understanding semantics. In fine-tuning mode, BERT is trained on the fake news dataset using a classification head. This allows it to adapt to the domain-specific language of news content.
The BERT architecture processes input text through multiple transformer blocks, learning complex features through self-attention mechanisms and residual connections.

5. Model Evaluation

After the model is trained, it must be evaluated to understand its effectiveness. Common evaluation metrics used in this context are:

- Accuracy: Measures the overall correctness of the model.
- Precision: Measures how many instances predicted as fake are actually fake.
- Recall: Measures how many of the actual fake news items were correctly detected.
- F1 Score: Harmonic mean of precision and recall, giving a balanced measure.

These metrics are calculated using a confusion matrix, which categorizes predictions into true positives, true negatives, false positives, and false negatives. A high precision and recall indicate a reliable model, especially important in scenarios where fake news must be detected with minimal false alarms.

6. Real-Time Prediction and Inference

Once trained, the model can be deployed for real-time inference. A prediction system typically involves:

- Accepting user input (e.g., a news article or headline).
- Applying the same preprocessing steps as used during training.
- Transforming the text into a feature vector or embedding.
- Using the trained model to classify the input as “fake” or “real”.

In deployment scenarios, this can be built into a web app, browser extension, or integrated with news feeds or social media APIs to flag suspicious content.

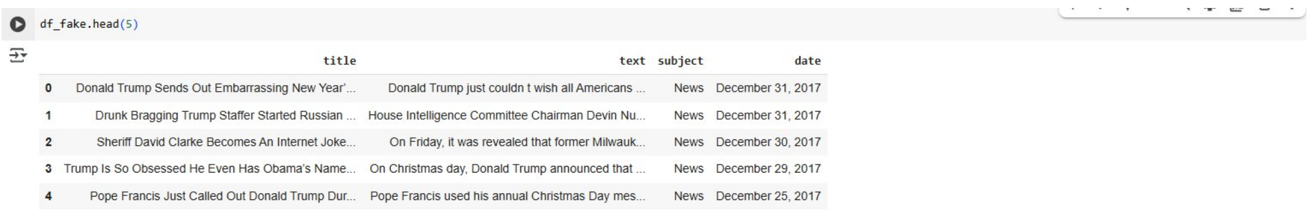
IMPLEMENTATION

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from sklearn.metrics import classification_report
import re
import string

df_fake = pd.read_csv("/content/Fake.csv")
df_true = pd.read_csv("/content/True.csv")
df_fake.head(5)
df_true.head(5)
df_fake["class"] = 0
df_true["class"] = 1
df_fake.shape, df_true.shape
df_fake_manual_testing = df_fake.tail(10)
for i in range(23480,23470,-1):
   df_fake.drop([i], axis = 0, inplace = True)
df_true_manual_testing = df_true.tail(10)
for i in range(21416,21406,-1):
   df_true.drop([i], axis = 0, inplace = True)
df_fake.shape, df_true.shape
df_fake_manual_testing.loc[:, "class"] = 0
df_fake_manual_testing.loc[:, "class"] = 1
df_fake_manual_testing.head(10)
df_true_manual_testing.head(10)
df_manual_testing = pd.concat([df_fake_manual_testing,df_true_manual_testing], axis = 0)
df_manual_testing.to_csv("manual_testing.csv")
df_marge = pd.concat([df_fake, df_true], axis =0 )
```

```
df_marge.head(10)
df_marge.columns
df = df_marge.drop(["title", "subject","date"], axis = 1)
df.isnull().sum()
df = df.sample(frac = 1)
df.head()
df.reset_index(inplace = True)
df.drop(["index"], axis = 1, inplace = True)
df.reset_index(inplace = True)
df.drop(["index"], axis = 1, inplace = True)
df.columns
df.head()
def wordopt(text):
    text = text.lower()
    text = re.sub('\[.*?\]', '', text)
    text = re.sub("\\W"," ",text)
    text = re.sub('https?://\S+|www\.\S+', '', text)
    text = re.sub('<.*?>+', '', text)
    text = re.sub('[%s]' % re.escape(string.punctuation), '', text)
    text = re.sub('\n', '', text)
    text = re.sub('\w*\d\w*', '', text)
    return text
df["text"] = df["text"].apply(wordopt)
x = df["text"]
y = df["class"]
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.25)

from sklearn.feature_extraction.text import TfidfVectorizer
vectorization = TfidfVectorizer()
xv_train = vectorization.fit_transform(x_train)
xv_test = vectorization.transform(x_test)
from sklearn.linear_model import LogisticRegression
LR = LogisticRegression()
LR.fit(xv_train,y_train)
```

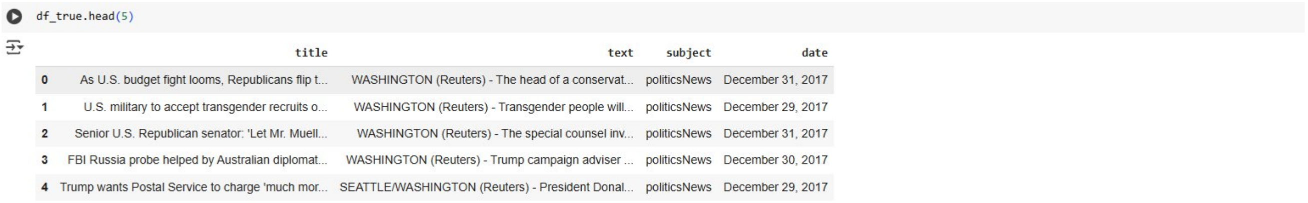
```
pred_lr=LR.predict(xv_test)
LR.score(xv_test, y_test
print(classification_report(y_test, pred_lr))

from sklearn.tree import DecisionTreeClassifier
DT = DecisionTreeClassifier()
DT.fit(xv_train, y_train)
pred_dt = DT.predict(xv_test)
DT.score(xv_test, y_test)
print(classification_report(y_test, pred_dt))

from sklearn.ensemble import GradientBoostingClassifier
GBC = GradientBoostingClassifier(random_state=0)
GBC.fit(xv_train, y_train)
pred_gbc = GBC.predict(xv_test)
GBC.score(xv_test, y_test)
print(classification_report(y_test, pred_gbc))

from sklearn.ensemble import RandomForestClassifier
RFC = RandomForestClassifier(random_state=0)
RFC.fit(xv_train, y_train)
pred_rfc = RFC.predict(xv_test)
RFC.score(xv_test, y_test)
print(classification_report(y_test, pred_rfc))
def output_lable(n):
    if n == 0:
        return "Fake News"
    elif n == 1:
        return "Not A Fake News"

def manual_testing(news):
    testing_news = {"text":[news]}
    new_def_test = pd.DataFrame(testing_news)
    new_def_test["text"] = new_def_test["text"].apply(wordopt)
    new_x_test = new_def_test["text"]
    new_xv_test = vectorization.transform(new_x_test)
    pred_LR = LR.predict(new_xv_test)
    pred_DT = DT.predict(new_xv_test)
    pred_GBC = GBC.predict(new_xv_test)
    pred_RFC = RFC.predict(new_xv_test)

    return print("\n\nLR Prediction: {} \nDT Prediction: {} \nGBC Prediction: {} \nRFC Prediction:
{}".format(output_lable(pred_LR[0]),
                                                                                                output_lable(pred_DT[0]),
                                                                                                output_lable(pred_GBC[0]),
                                                                                                output_lable(pred_RFC[0])))
news = str(input())
manual_testing(news)
```

OUTPUT



```
df_fake.head(5)
```

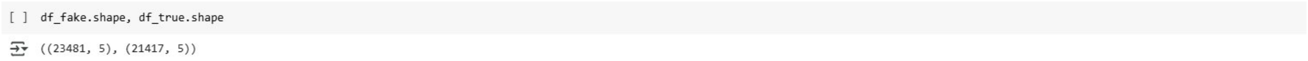
	title	text	subject	date
0	Donald Trump Sends Out Embarrassing New Year'...	Donald Trump just couldn t wish all Americans ...	News	December 31, 2017
1	Drunk Bragging Trump Staffer Started Russian ...	House Intelligence Committee Chairman Devin Nu...	News	December 31, 2017
2	Sheriff David Clarke Becomes An Internet Joke...	On Friday, it was revealed that former Milwauk...	News	December 30, 2017
3	Trump Is So Obsessed He Even Has Obama's Name...	On Christmas day, Donald Trump announced that ...	News	December 29, 2017
4	Pope Francis Just Called Out Donald Trump Dur...	Pope Francis used his annual Christmas Day mes...	News	December 25, 2017



```
df_true.head(5)
```

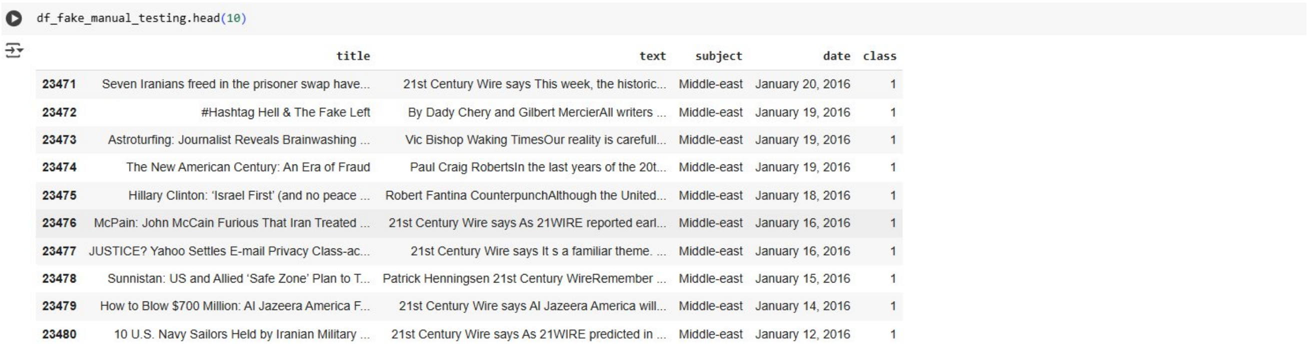
	title	text	subject	date
0	As U.S. budget fight looms, Republicans flip t...	WASHINGTON (Reuters) - The head of a conservat...	politicsNews	December 31, 2017
1	U.S. military to accept transgender recruits o...	WASHINGTON (Reuters) - Transgender people will...	politicsNews	December 29, 2017
2	Senior U.S. Republican senator: 'Let Mr. Muell...	WASHINGTON (Reuters) - The special counsel inv...	politicsNews	December 31, 2017
3	FBI Russia probe helped by Australian diplomat...	WASHINGTON (Reuters) - Trump campaign adviser ...	politicsNews	December 30, 2017
4	Trump wants Postal Service to charge 'much mor...	SEATTLE/WASHINGTON (Reuters) - President Donal...	politicsNews	December 29, 2017

Inserting a column called "class" for fake and real news dataset to categories fake and true news.



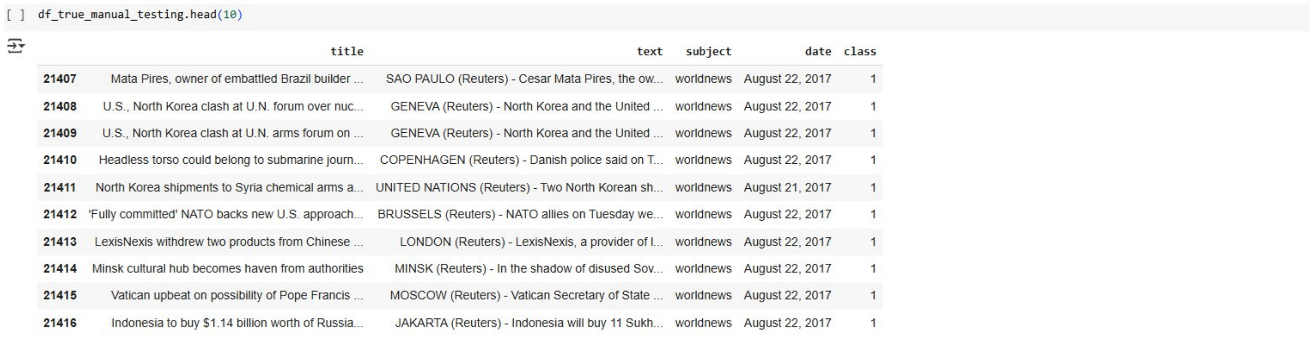
```
df_fake.shape, df_true.shape
```

```
((23481, 5), (21417, 5))
```



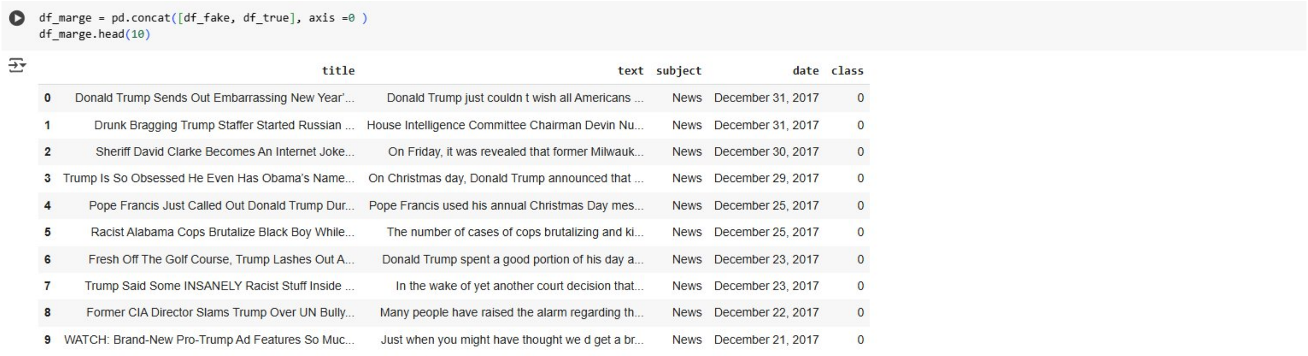
```
df_fake_manual_testing.head(10)
```

	title	text	subject	date	class
23471	Seven Iranians freed in the prisoner swap have...	21st Century Wire says This week, the historic...	Middle-east	January 20, 2016	1
23472	#Hashtag Hell & The Fake Left	By Dady Chery and Gilbert MercierAll writers ...	Middle-east	January 19, 2016	1
23473	Astroturfing: Journalist Reveals Brainwashing ...	Vic Bishop Waking TimesOur reality is carefull...	Middle-east	January 19, 2016	1
23474	The New American Century: An Era of Fraud	Paul Craig RobertsIn the last years of the 20t...	Middle-east	January 19, 2016	1
23475	Hillary Clinton: 'Israel First' (and no peace ...	Robert Fantina CounterpunchAlthough the United...	Middle-east	January 18, 2016	1
23476	McPain: John McCain Furious That Iran Treated ...	21st Century Wire says As 21WIRE reported earl...	Middle-east	January 16, 2016	1
23477	JUSTICE? Yahoo Settles E-mail Privacy Class-ac...	21st Century Wire says It s a familiar theme. ...	Middle-east	January 16, 2016	1
23478	Sunnistan: US and Allied 'Safe Zone' Plan to T...	Patrick Henningsen 21st Century WireRemember ...	Middle-east	January 15, 2016	1
23479	How to Blow $700 Million: Al Jazeera America F...	21st Century Wire says Al Jazeera America will...	Middle-east	January 14, 2016	1
23480	10 U.S. Navy Sailors Held by Iranian Military ...	21st Century Wire says As 21WIRE predicted in ...	Middle-east	January 12, 2016	1



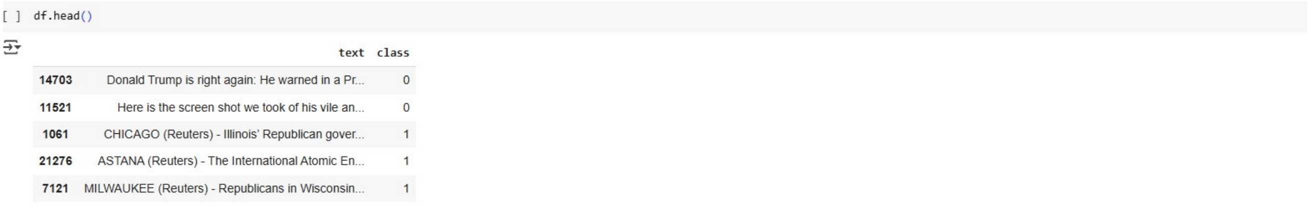
```
df_true_manual_testing.head(10)
```

	title	text	subject	date	class
21407	Mata Pires, owner of embattled Brazil builder ...	SAO PAULO (Reuters) - Cesar Mata Pires, the ow...	worldnews	August 22, 2017	1
21408	U.S., North Korea clash at U.N. forum over nuc...	GENEVA (Reuters) - North Korea and the United ...	worldnews	August 22, 2017	1
21409	U.S., North Korea clash at U.N. arms forum on ...	GENEVA (Reuters) - North Korea and the United ...	worldnews	August 22, 2017	1
21410	Headless torso could belong to submarine journ...	COPENHAGEN (Reuters) - Danish police said on T...	worldnews	August 22, 2017	1
21411	North Korea shipments to Syria chemical arms a...	UNITED NATIONS (Reuters) - Two North Korean sh...	worldnews	August 21, 2017	1
21412	'Fully committed' NATO backs new U.S. approach...	BRUSSELS (Reuters) - NATO allies on Tuesday we...	worldnews	August 22, 2017	1
21413	LexisNexis withdrew two products from Chinese ...	LONDON (Reuters) - LexisNexis, a provider of l...	worldnews	August 22, 2017	1
21414	Minsk cultural hub becomes haven from authorities	MINSK (Reuters) - In the shadow of disused Sov...	worldnews	August 22, 2017	1
21415	Vatican upbeat on possibility of Pope Francis ...	MOSCOW (Reuters) - Vatican Secretary of State ...	worldnews	August 22, 2017	1
21416	Indonesia to buy $1.14 billion worth of Russia...	JAKARTA (Reuters) - Indonesia will buy 11 Sukh...	worldnews	August 22, 2017	1



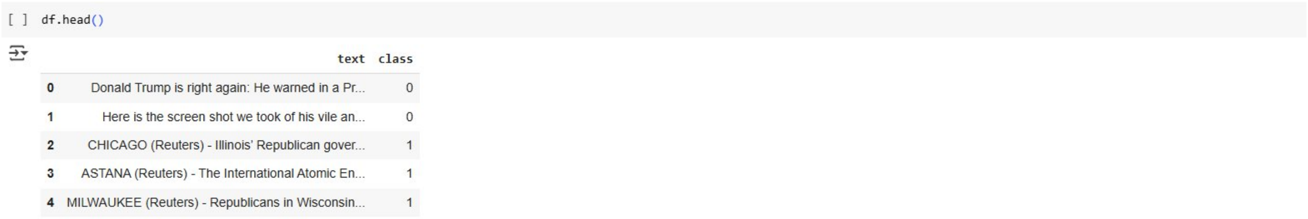
```
df_marge = pd.concat([df_fake, df_true], axis =0 )
df_marge.head(10)
```

	title	text	subject	date	class
0	Donald Trump Sends Out Embarrassing New Year'...	Donald Trump just couldn t wish all Americans ...	News	December 31, 2017	0
1	Drunk Bragging Trump Staffer Started Russian ...	House Intelligence Committee Chairman Devin Nu...	News	December 31, 2017	0
2	Sheriff David Clarke Becomes An Internet Joke...	On Friday, it was revealed that former Milwauk...	News	December 30, 2017	0
3	Trump Is So Obsessed He Even Has Obama's Name...	On Christmas day, Donald Trump announced that ...	News	December 29, 2017	0
4	Pope Francis Just Called Out Donald Trump Dur...	Pope Francis used his annual Christmas Day mes...	News	December 25, 2017	0
5	Racist Alabama Cops Brutalize Black Boy While...	The number of cases of cops brutalizing and ki...	News	December 25, 2017	0
6	Fresh Off The Golf Course, Trump Lashes Out A...	Donald Trump spent a good portion of his day a...	News	December 23, 2017	0
7	Trump Said Some INSANELY Racist Stuff Inside ...	In the wake of yet another court decision that...	News	December 23, 2017	0
8	Former CIA Director Slams Trump Over UN Bully...	Many people have raised the alarm regarding th...	News	December 22, 2017	0
9	WATCH: Brand-New Pro-Trump Ad Features So Muc...	Just when you might have thought we d get a br...	News	December 21, 2017	0



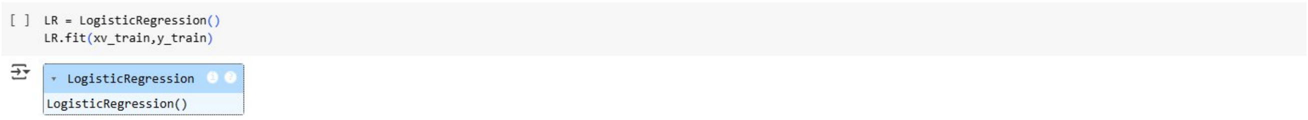
```
df.head()
```

	text	class
14703	Donald Trump is right again: He warned in a Pr...	0
11521	Here is the screen shot we took of his vile an...	0
1061	CHICAGO (Reuters) - Illinois' Republican gover...	1
21276	ASTANA (Reuters) - The International Atomic En...	1
7121	MILWAUKEE (Reuters) - Republicans in Wisconsin...	1



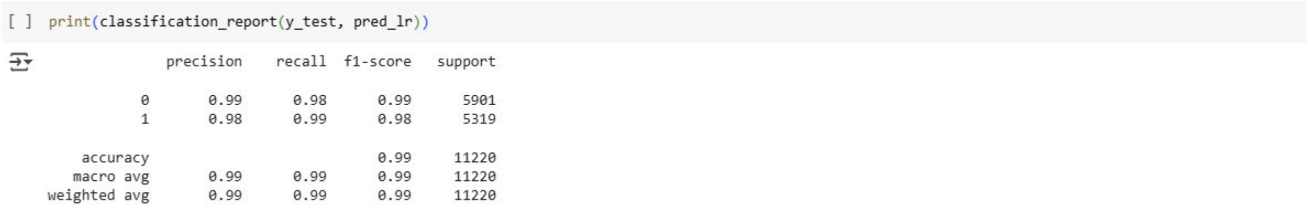
```
df.head()
```

	text	class
0	Donald Trump is right again: He warned in a Pr...	0
1	Here is the screen shot we took of his vile an...	0
2	CHICAGO (Reuters) - Illinois' Republican gover...	1
3	ASTANA (Reuters) - The International Atomic En...	1
4	MILWAUKEE (Reuters) - Republicans in Wisconsin...	1



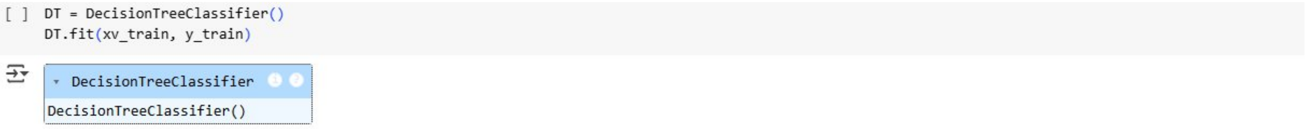
```
LR = LogisticRegression()
LR.fit(xv_train,y_train)
```

```
LogisticRegression
LogisticRegression()
```



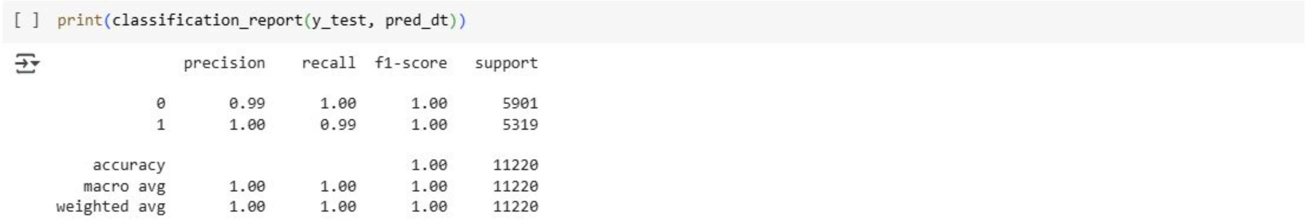
```
print(classification_report(y_test, pred_lr))
```

```
              precision    recall  f1-score   support

           0       0.99      0.98      0.99      5901
           1       0.98      0.99      0.98      5319

    accuracy                           0.99     11220
   macro avg       0.99      0.99      0.99     11220
weighted avg       0.99      0.99      0.99     11220
```



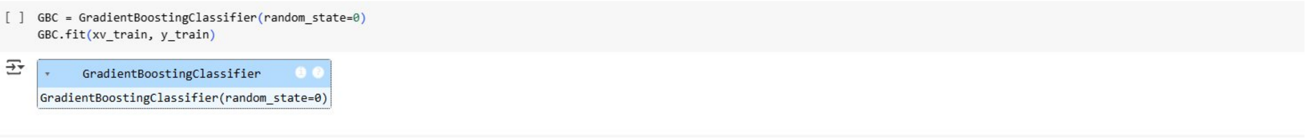
```
[ ] DT = DecisionTreeClassifier()
    DT.fit(xv_train, y_train)
```

```
DecisionTreeClassifier
DecisionTreeClassifier()
```



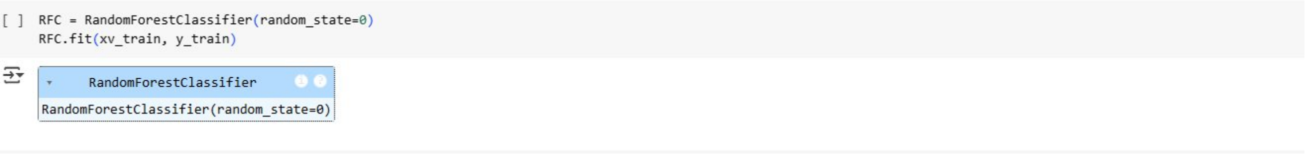
```
[ ] print(classification_report(y_test, pred_dt))
```

```
              precision    recall  f1-score   support

           0       0.99      1.00      1.00      5901
           1       1.00      0.99      1.00      5319

    accuracy                           1.00     11220
   macro avg       1.00      1.00      1.00     11220
weighted avg       1.00      1.00      1.00     11220
```



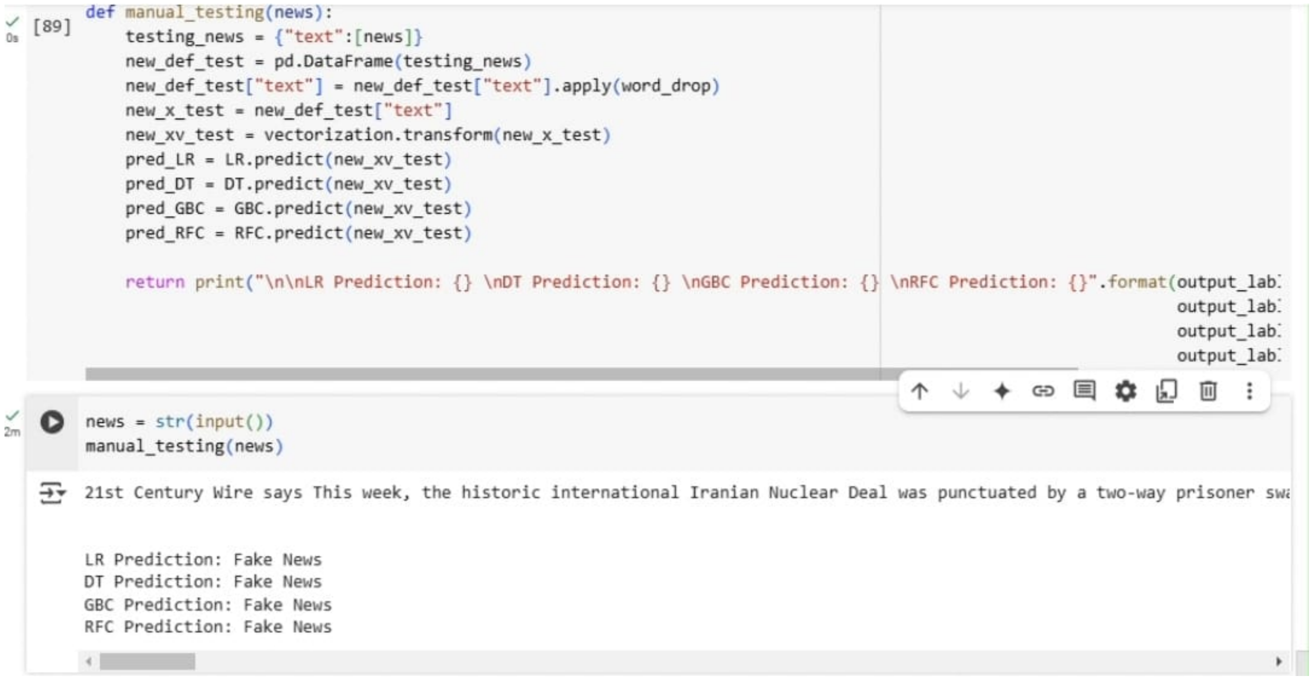
```
[ ] GBC = GradientBoostingClassifier(random_state=0)
    GBC.fit(xv_train, y_train)
```

```
GradientBoostingClassifier
GradientBoostingClassifier(random_state=0)
```



```
[ ] RFC = RandomForestClassifier(random_state=0)
    RFC.fit(xv_train, y_train)
```

```
RandomForestClassifier
RandomForestClassifier(random_state=0)
```



```
[89] def manual_testing(news):
         testing_news = {"text":[news]}
         new_def_test = pd.DataFrame(testing_news)
         new_def_test["text"] = new_def_test["text"].apply(word_drop)
         new_x_test = new_def_test["text"]
         new_xv_test = vectorization.transform(new_x_test)
         pred_LR = LR.predict(new_xv_test)
         pred_DT = DT.predict(new_xv_test)
         pred_GBC = GBC.predict(new_xv_test)
         pred_RFC = RFC.predict(new_xv_test)

         return print("\n\nLR Prediction: {} \nDT Prediction: {} \nGBC Prediction: {} \nRFC Prediction: {}".format(output_lab
                                                                                                                   output_lab
                                                                                                                   output_lab
                                                                                                                   output_lab
```

```
news = str(input())
manual_testing(news)
```

```
21st Century Wire says This week, the historic international Iranian Nuclear Deal was punctuated by a two-way prisoner sw


LR Prediction: Fake News
DT Prediction: Fake News
GBC Prediction: Fake News
RFC Prediction: Fake News
```

APPLICATIONS

In today's digital age, the rapid spread of misinformation has become a significant concern, impacting society, politics, health, and individual decision-making. Fake news detection systems serve as a critical technological solution to combat this challenge. These systems leverage Natural Language Processing (NLP), Machine Learning (ML), and Deep Learning (DL) techniques to automatically identify and filter false or misleading information across various platforms. The following are key applications of fake news detection technologies:

1. Social Media Monitoring: Social media platforms such as Facebook, Twitter, and Instagram are major sources of news consumption. Fake news detection systems can be integrated into these platforms to identify and block the spread of false content in real time, helping maintain information integrity and prevent mass misinformation campaigns.
2. News Aggregators and Portals: Online news platforms and aggregators can implement fake news detection tools to evaluate the credibility of submitted or syndicated articles. This ensures that only trustworthy content is promoted to readers, thereby increasing public confidence in digital journalism.
3. Government and Political Use: Governments and election commissions can use fake news detection systems during sensitive periods such as elections or referendums. These tools can monitor misinformation campaigns and mitigate their influence on public opinion and voting behavior.
4. Fact-checking Organizations: Fake news detection systems assist fact-checking websites like Snopes, PolitiFact, and Alt News in automating the initial screening process of content for review. This increases efficiency and helps prioritize the most impactful or viral pieces of misinformation.
5. Healthcare and Pandemic Response: During health crises (e.g., COVID-19 pandemic), fake news detection can prevent the spread of dangerous medical misinformation regarding treatments, vaccines, and safety measures, thus protecting public health and safety.
6. Education and Awareness: Educational institutions and research organizations can employ fake news detectors to teach students about digital literacy and the importance of verifying information. These tools can also be used to study misinformation trends and develop counter-strategies.

CONCLUSION

Fake news has emerged as a pervasive issue in the digital era, affecting individuals, communities, governments, and organizations worldwide. The proliferation of social media and online content-sharing platforms has made it easier than ever for misinformation to spread rapidly, often resulting in serious societal, political, and economic consequences. In this context, the development of a reliable and efficient fake news detection system becomes not only a technological challenge but also a social necessity.

This project demonstrates how a combination of Natural Language Processing and Machine Learning techniques can be employed to effectively identify fake news. Starting from data preprocessing and feature extraction to model training and evaluation, each phase of the system is crucial to building a robust and accurate classifier. By leveraging both traditional algorithms such as Logistic Regression and advanced models like LSTM and BERT, the system is capable of understanding not only the superficial structure of news articles but also the deeper semantics and contextual cues that differentiate fake content from real.

Furthermore, the project explores various real-world applications of fake news detection systems, highlighting their potential in media platforms, public health, politics, education, and corporate environments. The deployment of such systems can significantly reduce the impact of misinformation and contribute to a more informed and aware society.

In conclusion, while fake news detection is a challenging task due to the ever-changing nature of language and tactics used by misinformation spreaders, continuous advancements in machine learning and AI provide a promising path forward. Future work may include improving model accuracy through larger and more diverse datasets, enhancing multilingual capabilities, and integrating real-time user feedback. With persistent efforts and innovation, automated fake news detection systems can become an essential tool in preserving truth and transparency in the digital world.