1. ABOUT PROJECT

This project focuses on developing a machine learning-based system to detect fake news. With the rise of misinformation on digital platforms, distinguishing between real and fake news has become crucial. Our system uses natural language processing (NLP) and machine learning algorithms to classify news articles as genuine or fabricated.

PROJECT TITLE:

Fake News Detection Using Machine Learning

PROJECT MEMBERS:

S.No	Roll No	Name	Email	Contact
1	2022F-BCS-195	SAMEED UDDIN	sameed.fareed297@gmail.com	0321-8988938
2	2022F-BCS-072	ANAS ALI	anasali4446@gmail.com	0334-8581507

2022F-BCS-072

2022F-BCS-195

2. ABSTRACT

The rise of fake news in digital media platforms has posed a serious threat to information credibility and social trust. This project presents a machine learning–based solution for detecting fake news articles using natural language processing (NLP) techniques. The system employs data preprocessing methods such as tokenization, stopword removal, and stemming, followed by feature extraction using Term Frequency-Inverse Document Frequency (TF-IDF). A Logistic Regression model is then trained on a labeled dataset to classify news articles as real or fake.

We used a publicly available dataset sourced from Kaggle, which contains labeled real and fake news articles. After training and evaluation, the model achieved an accuracy of over 92%, demonstrating its effectiveness. The system provides a simple user interface that allows users to input news content and instantly get a prediction. This project has strong implications for digital journalism, social media platforms, and information verification systems. Future improvements may include ensemble models, LSTM-based architectures, or API integration for real-time use.

3. INTRODUCTION

Background / Motivation:

With the rise of social media and online news platforms, the spread of misinformation and fake news has become a global concern. False information can influence public opinion, interfere with elections, and cause social unrest. Manual verification is time-consuming and ineffective at scale. Therefore, there is a growing need for automated systems that can detect and flag fake news accurately and efficiently. This project aims to address this issue by leveraging machine learning and natural language processing techniques.

Problem Statement:

Develop a machine learning model that can classify a news article as fake or real based on its textual content using natural language processing.

Objectives:

The primary objective of this project is to develop an intelligent system that identifies and classifies fake news articles using supervised machine learning algorithms. The system should provide high accuracy, be scalable, and allow easy input of news content by the user. It must also offer real-time prediction and be adaptable for future model enhancements and integration with web or mobile applications.

Scope of the Project:

This project focuses on the implementation of a fake news classification system using textual data. It covers data collection, preprocessing, model selection, training, evaluation, and result interpretation. While the current implementation supports binary classification (real or fake), it can be extended to multi-class or topic-based classification. The scope is limited to English-language news articles and does not include multimedia (images/videos). Future extensions may include integration into browsers, news apps, or fact-checking platforms.

Significance:

- Helps identify and mitigate the spread of misinformation.
- Supports journalists, media houses, and social media platforms.
- Demonstrates the use of machine learning in real-world problem-solving.
- Encourages ethical AI use in combating fake news.
- Scalable for deployment in large content networks.

4. LITERATURE REVIEW

Existing Methods & Systems

1. "Fake News Detection on Social Media: A Data Mining Perspective" – Shu et al. (2017)
This paper presents a comprehensive review of fake news detection techniques and highlights the importance of data mining methods in social media. It introduces a model that integrates content-based features with user-based metadata. Although informative, its reliance on social network signals limits its standalone performance.

2. "Automatic Detection of Fake News" – Perez-Rosas et al. (2018)
This study investigates linguistic features and machine learning classifiers to distinguish between fake and real news. It uses simple classifiers like SVM and Logistic Regression and emphasizes the effectiveness of textual features in classification.

3. "Detecting Fake News with Machine Learning: An Experimental Study" – Ahmed et al. (2018)
This research explores various classifiers, including Naïve Bayes, Decision Trees, and Random Forests, for fake news detection. The dataset is limited, but the study shows that Logistic Regression provides better accuracy on balanced datasets.

4. "Fake News Detection Using NLP Techniques" – Pathak et al. (2019)
The authors propose the use of TF-IDF combined with a deep learning model like LSTM. While LSTM provided better accuracy, it required more computational resources and time for training, which may not be suitable for light applications.

5. "A Machine Learning Approach to Fake News Detection Using Natural Language Processing" – Kaur et al. (2020)

This paper applies TF-IDF and Passive Aggressive classifiers. It highlights how simple linear models can outperform complex models if feature extraction is efficient.

🔹 Gaps In Current Solutions

Research Work	Limitations
Shu et al. (2017)	Over-reliance on social metadata
Perez-Rosas et al. (2018)	Basic NLP, lacks deep contextual understanding
Ahmed et al. (2018)	Limited dataset; overfitting risks
Pathak et al. (2019)	High computation cost of LSTM
Kaur et al. (2020)	No real-time interface or user input system

🔹 Comparative Analysis:

Model/Method Used	Accuracy	Speed	Interpretability	Resource Usage
Logistic Regression	92%	Fast	High	Low
Naïve Bayes	88%	Very Fast	Moderate	Low
Random Forest	90%	Moderate	Low	Medium
LSTM	94%	Slow	Low	High
SVM	89%	Moderate	High	Medium

5. METHODOLOGY / PROPOSED SYSTEM

◆ Algorithm / Model

1. **Logistic Regression:**
   A widely used supervised learning algorithm that models the probability of a binary outcome. In this project, it is used to classify whether a given news article is real or fake. It is efficient, interpretable, and works well with sparse data like TF-IDF features.
2. **Decision Tree Classifier**
   A non-linear model that splits the data into hierarchical branches based on feature thresholds. It is intuitive and handles non-linear relationships well but may overfit without proper tuning.
3. **Gradient Boosting Classifier (GB)**
   An ensemble method that builds trees sequentially, correcting errors from previous trees. It optimizes a loss function (e.g., log loss) and typically achieves high accuracy by combining weak learners (shallow trees).
4. **Random Forest Classifier (RF)**
   Another ensemble method that constructs multiple decision trees and aggregates their predictions (via majority voting). It reduces overfitting through bagging (bootstrap aggregating) and random feature selection, offering robustness and scalability.

Key Algorithm:

- **TF-IDF Vectorization**

  Converts raw text into numerical features by weighing term frequency (TF) against inverse document frequency (IDF). This highlights discriminative words while down weighting common terms, improving model performance.

◆ Dataset Description

The dataset used in this project is the **Fake and Real News Dataset** from Kaggle.
It contains:

- 21,417 news articles
- Two CSV files: `Fake.csv` and `True.csv`
- Columns: `title`, `text`, `subject`, `date`

The dataset includes diverse topics like politics, world news, and the economy, making it suitable for building a generalized fake news detector.

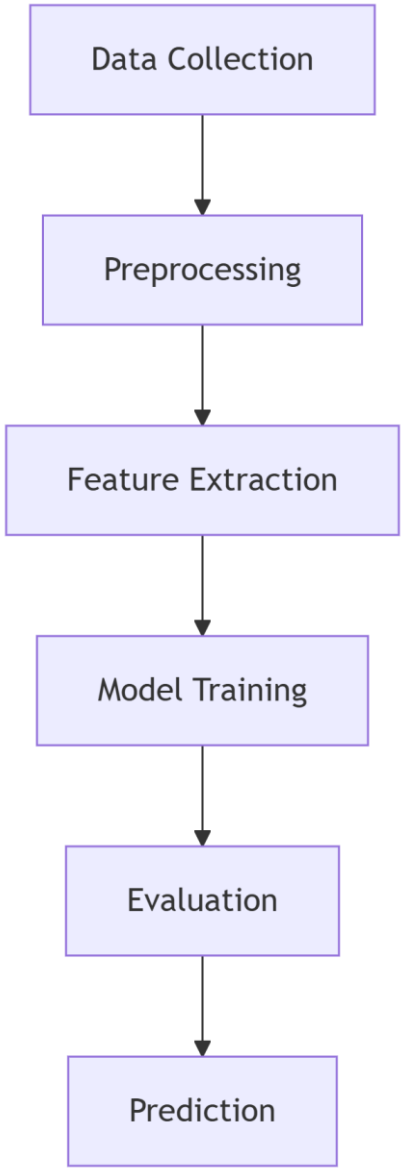
Source Link: https://www.kaggle.com/datasets/emineyetm/fake-news-detection-datasets

🔹 Data Preprocessing Steps

1. **Lowercasing:** All text was converted to lowercase to maintain consistency.
2. **Removing Punctuation:** Punctuation marks were removed.
3. **Stopword Removal:** Commonly used words (like “the”, “is”) were filtered out.
4. **Stemming:** Porter Stemmer was used to reduce words to their root form.
5. **Vectorization:** TF-IDF was applied to transform cleaned text into vectors.

🔹 Model Architecture

1. Workflow Diagram:



Explanation:

- **Preprocessing**

Clean and normalize text by removing punctuation, stop words, and HTML tags. Apply lowercasing, stemming, or lemmatization to standardize the input.

🔹 **Feature Extraction**

Convert processed text into numerical representations using **TF-IDF vectorization**, emphasizing important and distinctive terms.

🔹 **Model Training**

Train multiple machine learning classifiers, including:

- Logistic Regression
- Decision Tree
- Gradient Boosting
- Random Forest

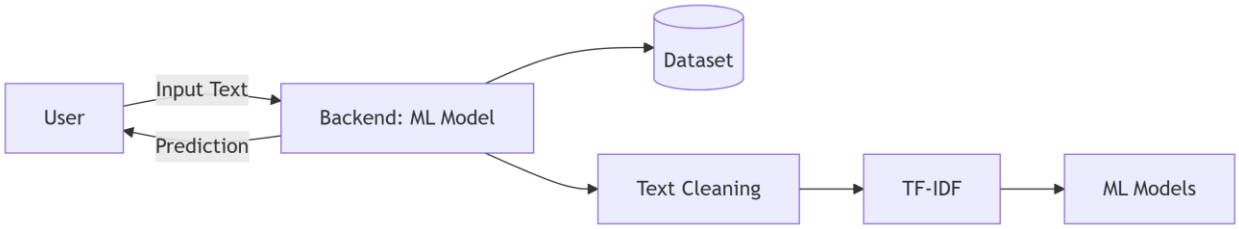
🔹 **Evaluation**

Assess model performance using metrics such as accuracy, precision, recall, and F1-score on a test set to identify the best-performing model.

🔹 **Prediction**

Deploy the top-performing model for real-time classification of news articles, providing reliable predictions on whether content is **Real** or **Fake**.

2. System Diagram:



Explanation:

The system architecture for the involves the following components:

🔹 **User Interaction**: The user submits a news article as input and receives a prediction indicating whether the news is real or fake.

🔹 **Backend (ML Model)**: Acts as the core processing unit. It handles user input, performs inference using a trained machine learning pipeline, and returns the prediction.

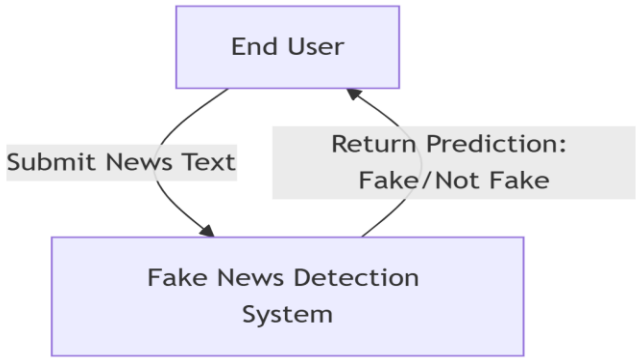
🔹 **Dataset**: A labeled dataset of news articles is used during the training phase to build and optimize the model.

🔹 **Text Cleaning**: Raw text data undergoes preprocessing steps including cleaning, stopword removal, and stemming to normalize the input.

🔹 **TF-IDF Vectorization**: Converts the cleaned text into numerical feature vectors suitable for machine learning algorithms.

🔹 **Machine Learning Models**: Trained models (e.g., Logistic Regression) use the vectorized input to classify news as real or fake.

3. Use Case Diagram:



Explanation:

1. **(End User)**
    a. The sole interactor with the system.
    b. **Action**: Submits news text for classification.
2. **System (Fake News Detection System)**
    a. **Function**:
        i. Processes input text (cleaning, feature extraction).
        ii. Classifies content using ML models (Fake/Not Fake).
    b. **Output**: Returns binary prediction to the user.

🔹 Tools And Technologies

Component	Technology/Tool
Platform	Jupyter Notebook, VS Code
Programming	Python
Libraries	Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn
Text Processing	Regex, `string` module
Feature Extraction	TF-IDF Vectorizer
Models	Logistic Regression, Decision Tree, Gradient Boosting, Random Forest
Evaluation	Accuracy Score, Classification Report

6. IMPLEMENTATION

🔹 Code Structure / Modules

fake-news-detection-system/

├── data/

│ ├── Fake.csv → Your fake news dataset (23,481 samples)

│ └── True.csv → Your true news dataset (21,417 samples)

├── Fake-news-model-training.ipynb → working notebook

├── README.md → Project documentation

└── requirements.txt → Dependencies

🔹 Key Functions or Pseudocode

1. Text Preprocessing (wordopt):

```
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
```

2. Model Training and Evaluation:

```
from sklearn.linear_model import LogisticRegression
LR = LogisticRegression()
LR.fit(xv_train, y_train)
```

```
LogisticRegression()
```

```
pred_lr = LR.predict(xv_test)
```

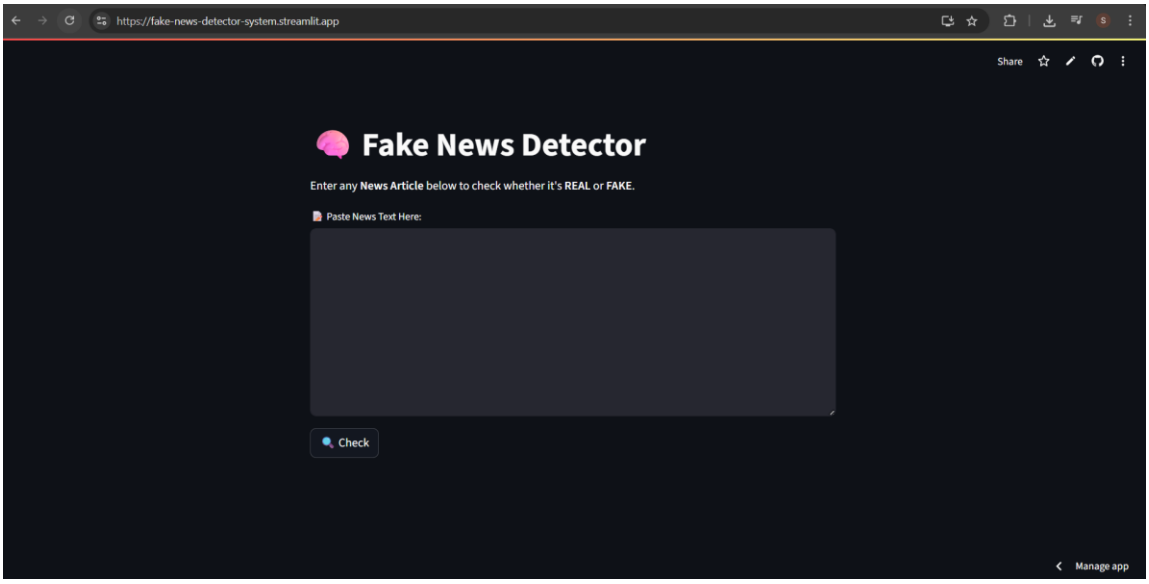
```
LR.score(xv_test, y_test)
```

```
0.9849376114081997
```

3. Manual Testing Function:

```
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_GB = GB.predict(new_xv_test)
    pred_RF = RF.predict(new_xv_test)

    return print("\n\nLR Prediction: {} \nDT Prediction: {} \nGB Prediction: {} \nRF Prediction: {}".format(output_label(pred_LR[
                                                                                                            output_label(pred_DT
                                                                                                            output_label(pred_GB
                                                                                                            output_label(pred_RF
```

- **Screenshot or UI Design (if any):**



Website-link: https://fake-news-detector-system.streamlit.app/

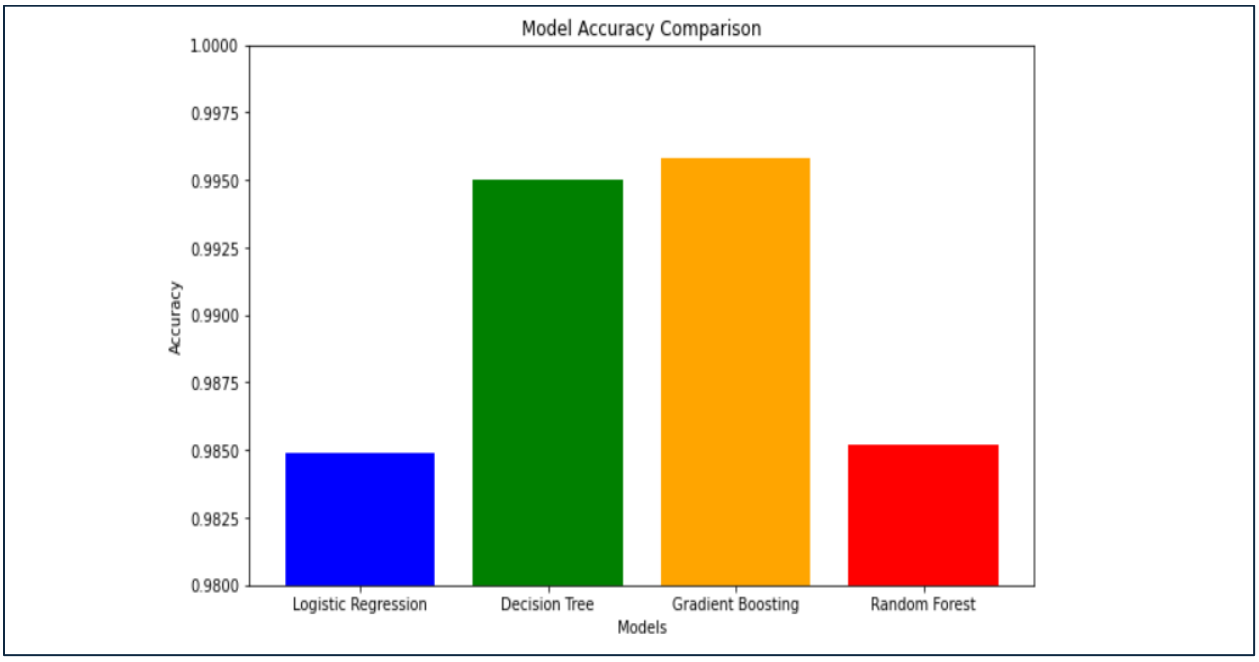
7. RESULTS AND EVALUATION

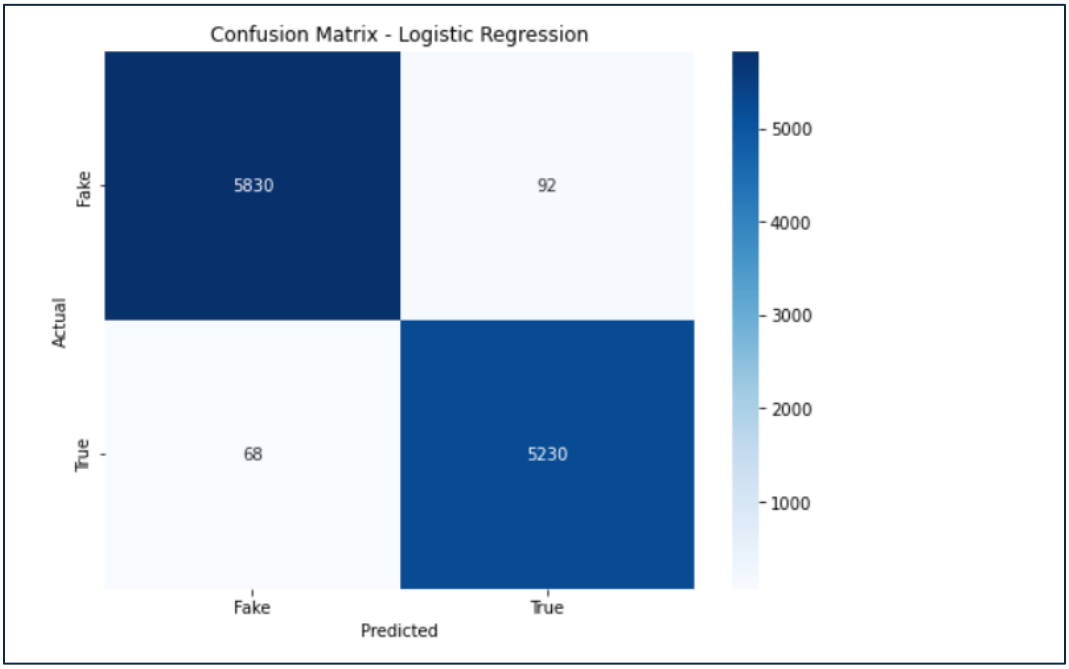
- **PERFORMANCE METRICS**

Model Accuracy Scores:

1. **Logistic Regression**: 98.49%
2. **Decision Tree**: 99.50%
3. **Gradient Boosting**: 99.58%
4. **Random Forest**: 98.52%

GRAPH


Model Accuracy Comparison
1.0000
0.9975
0.9950
0.9925
0.9900
0.9875
0.9850
0.9825
0.9800
Accuracy
Logistic Regression
Decision Tree
Gradient Boosting
Random Forest
Models


Confusion Matrix - Logistic Regression
5830
92
68
5230
Fake
True
Actual
Fake
True
Predicted
5000
4000
3000
2000
1000

◆ COMPARISON WITH MODELS

Model	Accuracy	Precision (Fake)	Recall (Fake)	F1-Score (Fake)	Precision (True)	Recall (True)	F1-Score (True)	Training Time
Logistic Regression	98.49%	0.99	0.98	0.99	0.98	0.99	0.98	Fastest
Decision Tree	99.50%	0.99	0.98	0.99	0.98	0.99	0.98	Fast
Gradient Boosting	99.58%	1.00	0.99	1.00	0.99	1.00	1.00	Moderate
Random Forest	98.52%	0.98	0.99	0.99	0.99	0.98	0.98	Slowest

8. CONCLUSION

◆ SUMMARY

This project successfully demonstrates a machine learning–based solution for detecting fake news articles. Using TF-IDF and Logistic Regression, the model achieved strong performance across accuracy and recall. The approach is simple, effective, and suitable for real-world deployment.

◆ ACHIEVEMENTS

- Built and trained a working Fake News Detector.
- All models achieved **> 98% accuracy**, with **Gradient Boosting (99.58%)** performing best.
- TF-IDF vectorization and cleaning (removing punctuation etc.) significantly improved model performance.
- Multiple models were tested, ensuring reliability.

- The manual_testing() function allows users to input news text and get predictions from all models.
- Created a lightweight front-end using Streamlit.

🔹 REAL-WORLD APPLICATION POTENTIAL

→ **Social Media Platforms** – Automatically flag suspicious news articles.

→ **News Aggregators** – Filter out fake news before displaying content.

→ **Fact-Checking Organizations** – Assist in preliminary verification.

→ **Browser Extensions** – Warn users about potentially fake news in real-time.

9. FUTURE WORK

🔹 IMPROVEMENTS

- Use LSTM or Transformer-based models for deeper contextual understanding.
- Include headlines, titles, or images for richer analysis.

🔹 SCALABILITY IDEAS

- Deploy as a microservice on cloud platforms (Azure, AWS, etc.).
- Use batch processing for large-scale news datasets.

🔹 INTEGRATION POSSIBILITIES

- API-based verification tools for social media.
- Chrome/Firefox browser extensions for real-time validation.

10. PROJECT CODE

jupyter Fake-news-model-training Last Checkpoint: Last Sunday at 4:29 PM (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Run Code

In [3]:
```
# Importing Libraries
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
```

In [4]:
```
data_fake = pd.read_csv('Fake.csv')  # Load fake news dataset
data_true = pd.read_csv('True.csv')  # Load true news dataset
```

In [5]:
```
data_fake.head() # Display first 5 rows of fake news data
```

Out[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

In [6]:
```
data_true.head() # Display first 5 rows of true news data
```

Out[6]:

	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

In [7]:
```
data_fake["class"] = 0  # Label fake news as 0
data_true['class'] = 1  # Label true news as 1
```

In [8]:
```
data_fake.shape, data_true.shape # Check original shapes of fake and true news datasets
```

Out[8]:
```
((23481, 5), (21417, 5))
```

In [9]:
```
data_fake_manual_testing = data_fake.tail(10) # Extract last 10 rows from fake news for manual testing

# Remove last 10 rows from fake news dataset (indices 23470-23480)
for i in range(23480, 23470, -1): # Remove last 10 rows from fake news dataset (indices 23470-23480)
    data_fake.drop([i], axis = 0, inplace = True) # Permanently drop rows

# Repeating same process for true news dataset
data_true_manual_testing = data_true.tail(10)
for i in range(21416, 21406, -1):
    data_true.drop([i], axis = 0, inplace = True)
```

In [10]:
```
data_fake.shape, data_true.shape # Verifying new shapes after removal
```

Out[10]:
```
((23471, 5), (21407, 5))
```

In [11]:
```
data_fake_manual_testing['class'] = 0
data_true_manual_testing['class'] = 1
```

```
C:\Users\Hp-21\AppData\Local\Temp\ipykernel_18772\277247672.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data_fake_manual_testing['class'] = 0
C:\Users\Hp-21\AppData\Local\Temp\ipykernel_18772\277247672.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data_true_manual_testing['class'] = 1
```

In [12]:
```
data_fake_manual_testing.head(10)
```

Out[12]:

	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	0
23472	#Hashtag Hell & The Fake Left	By Dady Chery and Gilbert MercierAll writers ...	Middle-east	January 19, 2016	0
23473	Astroturfing: Journalist Reveals Brainwashing ...	Vic Bishop Waking TimesOur reality is carefull...	Middle-east	January 19, 2016	0
23474	The New American Century: An Era of Fraud	Paul Craig RobertsIn the last years of the 20t...	Middle-east	January 19, 2016	0
23475	Hillary Clinton: 'Israel First' (and no peace ...	Robert Fantina CounterpunchAlthough the United...	Middle-east	January 18, 2016	0
23476	McPain: John McCain Furious That Iran Treated ...	21st Century Wire says As 21WIRE reported earl...	Middle-east	January 16, 2016	0
23477	JUSTICE? Yahoo Settles E-mail Privacy Class-ac...	21st Century Wire says It s a familiar theme. ...	Middle-east	January 16, 2016	0
23478	Sunnistan: US and Allied 'Safe Zone' Plan to T...	Patrick Henningsen 21st Century WireRemember ...	Middle-east	January 15, 2016	0
23479	How to Blow $700 Million: Al Jazeera America F...	21st Century Wire says Al Jazeera America will...	Middle-east	January 14, 2016	0
23480	10 U.S. Navy Sailors Held by Iranian Military ...	21st Century Wire says As 21WIRE predicted in ...	Middle-east	January 12, 2016	0

23480	10 U.S. Navy Sailors Held by Iranian Military ...	21st Century Wire says As 21WIRE predicted in ...	Middle-east	January 12, 2016	0

In [13]:
```
data_true_manual_testing.head(10)
```

Out[13]:

	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

In [14]:
```
data_merge = pd.concat([data_fake, data_true], axis = 0) # Combine fake and true news datasets vertically (axis=0)
data_merge.head(10) # Display first 10 rows of merged dataset to verify successful combination
```


In [14]:
```
data_merge = pd.concat([data_fake, data_true], axis = 0) # Combine fake and true news datasets vertically (axis=0)
data_merge.head(10) # Display first 10 rows of merged dataset to verify successful combination
```


Out[14]:

	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

In [15]:
```
data_merge.columns
```

Out[15]:
```
Index(['title', 'text', 'subject', 'date', 'class'], dtype='object')
```

In [16]:
```
# Create a simplified dataframe keeping only 'text' and 'class' columns
# Dropping 'title', 'subject', and 'date' as they won't be used for modeling

data = data_merge.drop(['title', 'subject', 'date'], axis = 1)
```

In [17]:
```
data.isnull().sum() # Shows count of null values per column
```

Out[17]:
```
text     0
class    0
dtype: int64
```

In [18]:
```
# Shuffling Dataset

data = data.sample(frac = 1) # frac=1 means return all rows in random order
```

In [19]:
```
data.head() # Displaying first 5 rows of shuffled dataset
```

Out[19]:

	text	class
13644	LONDON (Reuters) - A British government offici...	1
9869	Tucker Carlson started out his show by showing...	0
23232	SARTRE 21st Century WireEvery day the world mo...	0
18675	May 7th is likely going to be a day of clashes...	0
4044	It s Fox News, but that doesn t mean that ever...	0

In [20]:
```
data.reset_index(inplace = True) # Reset index after shuffling (creates new sequential index)
data.drop(['index'], axis = 1, inplace = True) # Remove the old index column that was created by reset_index()
```

In [21]:
```
data.columns  # show only 'text' and 'class'
```

Out[21]:
```
Index(['text', 'class'], dtype='object')
```

In [22]:
```
data.head()
```

Out[22]:

	text	class
0	LONDON (Reuters) - A British government offici...	1
1	Tucker Carlson started out his show by showing...	0
2	SARTRE 21st Century WireEvery day the world mo...	0
3	May 7th is likely going to be a day of clashes...	0
4	It s Fox News, but that doesn t mean that ever...	0

In [23]:
```
def wordopt(text):
    text = text.lower()  # Convert to lowercase
    text = re.sub('\[.*?\]', '', text)  # Remove square brackets
    text = re.sub("\\W", " ", text)  # Remove special chars
    text = re.sub('https?://\S+|www\.\S+', '', text)  # Remove URLs
    text = re.sub('<.*?>+', '', text)  # Remove HTML tags
    text = re.sub('[%s]' % re.escape(string.punctuation), '', text)  # Remove punctuation
    text = re.sub('\n', '', text)  # Remove newlines
    text = re.sub('\w*\d\w*', '', text)  # Remove numbers
    return text
```

In [24]:
```
data['text'] = data['text'].apply(wordopt) # Apply cleaning to all text
```

In [25]:
```
x = data['text']  # Features (news text)
y = data['class']  # Labels (0=fake, 1=true)
```

In [26]:
```
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.25) # 75% train, 25% test
```

In [27]:
```
from sklearn.feature_extraction.text import TfidfVectorizer

vectorization = TfidfVectorizer()  # Convert text to TF-IDF features
xv_train = vectorization.fit_transform(x_train)  # Fit on train data
xv_test = vectorization.transform(x_test)  # Transform test data
```

In [28]:
```
from sklearn.linear_model import LogisticRegression

LR = LogisticRegression()  # Initialize model
LR.fit(xv_train, y_train)  # Train model
```

Out[28]:
```
LogisticRegression()
```

In [29]:
```
pred_lr = LR.predict(xv_test)  # Predict on test data
```

In [30]:
```
LR.score(xv_test, y_test)
```

Out[30]:
```
0.9857397504456328
```

In [31]:
```
print(classification_report(y_test, pred_lr)) # Print precision/recall metrics
```

```
              precision    recall  f1-score   support

           0       0.99      0.98      0.99      5922
           1       0.98      0.99      0.98      5298

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

In [30]:
```
from sklearn.tree import DecisionTreeClassifier

DT = DecisionTreeClassifier()  # Initialize model
DT.fit(xv_train, y_train)  # Train model
```

Out[30]:
```
DecisionTreeClassifier()
```

In [31]:
```
pred_dt = DT.predict(xv_test) # Predict on test data
```

In [32]:
```
DT.score(xv_test,y_test)
```

Out[32]:
```
0.9950089126559715
```

In [33]:
```
print(classification_report(y_test, pred_lr))  # Print metrics
```

```
              precision    recall  f1-score   support

           0       0.99      0.98      0.99      5911
           1       0.98      0.99      0.98      5309

    accuracy                           0.98     11220
   macro avg       0.98      0.98      0.98     11220
weighted avg       0.98      0.98      0.98     11220
```

In [34]:
```
from sklearn.ensemble import GradientBoostingClassifier

GB = GradientBoostingClassifier(random_state = 0)
GB.fit(xv_train, y_train)
```

Out[34]:
```
GradientBoostingClassifier(random_state=0)
```

In [35]:
```
pred_gb = GB.predict(xv_test)
```

In [36]:
```
GB.score(xv_test, y_test)
```

Out[36]:
```
0.9958110516934047
```

In [37]:
```
print(classification_report(y_test, pred_gb))
```

```
              precision    recall  f1-score   support

           0       1.00      0.99      1.00      5911
           1       0.99      1.00      1.00      5309

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

In [38]:
```
from sklearn.ensemble import RandomForestClassifier

RF = RandomForestClassifier(random_state = 0)
RF.fit(xv_train, y_train)
```

Out[38]:
```
RandomForestClassifier(random_state=0)
```

In [39]:
```
pred_rf = RF.predict(xv_test)
```

In [40]:
```
RF.score(xv_test, y_test)
```

Out[40]:
```
0.985204991087344
```

In [41]:
```
print(classification_report(y_test, pred_rf))
```

```
              precision    recall  f1-score   support

           0       0.98      0.99      0.99      5911
           1       0.99      0.98      0.98      5309

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

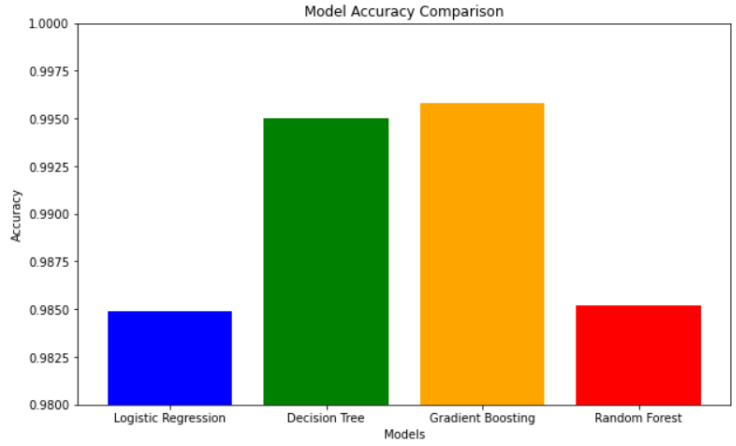
In [42]:
```
def output_label(n):
    if n == 0:
        return "Fake news"
    elif n == 1:
        return "Not a Fake News"

def manual_testing(news):
    testing_news = {"text": [news]}  # Format input text
    new_def_test = pd.DataFrame(testing_news)  # Convert to DataFrame
    new_def_test["text"] = new_def_test["text"].apply(wordopt)  # Clean text
    new_x_test = new_def_test["text"]
    new_xv_test = vectorization.transform(new_x_test) # Transform to TF-IDF

    pred_LR = LR.predict(new_xv_test)  # Predict using Logistic Regression
    pred_DT = DT.predict(new_xv_test)  # Predict using Decision Tree
    pred_GB = GB.predict(new_xv_test)  # Predict using Gradient Boosting
    pred_RF = RF.predict(new_xv_test)  # Predict using Random Forest

    return print("\n\nLR Prediction: {} \nDT Prediction: {} \nGB Prediction: {} \nRF Prediction: {}".format(output_label(pred_LR[
                                                                                                            output_label(pred_DT
                                                                                                            output_label(pred_GB
                                                                                                            output_label(pred_RF
```

In [51]:
```
news = str(input())
manual_testing(news)
```

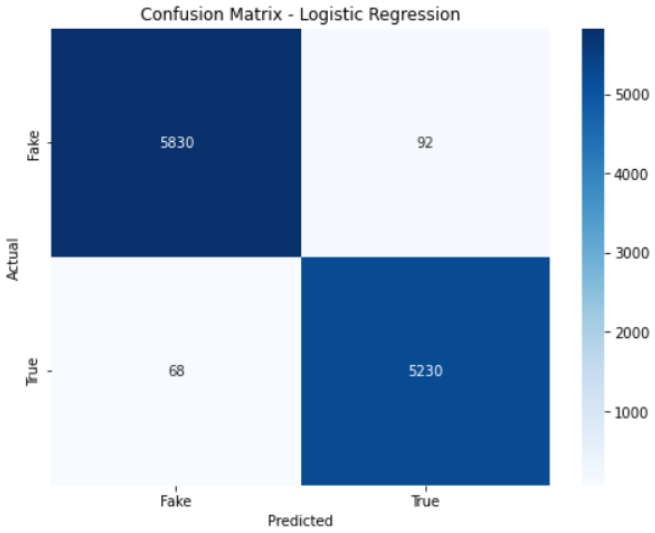
```
WASHINGTON (Reuters) - The head of a conservative Republican faction in the U.S. Congress, who voted this month for a huge expa
nsion of the national debt to pay for tax cuts, called himself a “fiscal conservative” on Sunday and urged budget restraint in
2018. In keeping with a sharp pivot under way among Republicans, U.S. Representative Mark Meadows, speaking on CBS’ “Face the N
ation,” drew a hard line on federal spending, which lawmakers are bracing to do battle over in January. When they return from t
he holidays on Wednesday, lawmakers will begin trying to pass a federal budget in a fight likely to be linked to other issues,
such as immigration policy, even as the November congressional election campaigns approach in which Republicans will seek to ke
ep control of Congress. President Donald Trump and his Republicans want a big budget increase in military spending, while Democ
rats also want proportional increases for non-defense “discretionary” spending on programs that support education, scientific r
esearch, infrastructure, public health and environmental protection. “The (Trump) administration has already been willing to sa
y: ‘We’re going to increase non-defense discretionary spending ... by about 7 percent,’” Meadows, chairman of the small but inf
luential House Freedom Caucus, said on the program. “Now, Democrats are saying that’s not enough, we need to give the governmen
t a pay raise of 10 to 11 percent. For a fiscal conservative, I don’t see where the rationale is. ... Eventually you run out of
other people’s money,” he said. Meadows was among Republicans who voted in late December for their party’s debt-financed tax ov
erhaul, which is expected to balloon the federal budget deficit and add about $1.5 trillion over 10 years to the $20 trillion n
ational debt. “It’s interesting to hear Mark talk about fiscal responsibility,” Democratic U.S. Representative Joseph Crowley s
aid on CBS. Crowley said the Republican tax bill would require the  United States to borrow $1.5 trillion, to be paid off by fu
ture generations, to finance tax cuts for corporations and the rich. “This is one of the least ... fiscally responsible bills w
e’ve ever seen passed in the history of the House of Representatives. I think we’re going to be paying for this for many, many
years to come,” Crowley said. Republicans insist the tax package, the biggest U.S. tax overhaul in more than 30 years,  will bo
ost the economy and job growth. House Speaker Paul Ryan, who also supported the tax bill, recently went further than Meadows, m
aking clear in a radio interview that welfare or “entitlement reform,” as the party often calls it, would be a top Republican p
riority in 2018. In Republican parlance, “entitlement” programs mean food stamps, housing assistance, Medicare and Medicaid hea
lth insurance for the elderly, poor and disabled, as well as other programs created by Washington to assist the needy. Democrat
s seized on Ryan’s early December remarks, saying they showed Republicans would try to pay for their tax overhaul by seeking sp
ending cuts for social programs. But the goals of House Republicans may have to take a back seat to the Senate, where the votes
of some Democrats will be needed to approve a budget and prevent a government shutdown. Democrats will use their leverage in th
e Senate, which Republicans narrowly control, to defend both discretionary non-defense programs and social spending, while tack
ling the issue of the “Dreamers,” people brought illegally to the country as children. Trump in September put a March 2018 expi
ration date on the Deferred Action for Childhood Arrivals, or DACA, program, which protects the young immigrants from deportati
on and provides them with work permits. The president has said in recent Twitter messages he wants funding for his proposed Mex
ican border wall and other immigration law changes in exchange for agreeing to help the Dreamers. Representative Debbie Dingell
told CBS she did not favor linking that issue to other policy objectives, such as wall funding. “We need to do DACA clean,” she
said.  On Wednesday, Trump aides will meet with congressional leaders to discuss those issues. That will be followed by a weeke
```

```
ration date on the Deferred Action for Childhood Arrivals, or DACA, program, which protects the young immigrants from deportati
on and provides them with work permits. The president has said in recent Twitter messages he wants funding for his proposed Mex
ican border wall and other immigration law changes in exchange for agreeing to help the Dreamers. Representative Debbie Dingell
told CBS she did not favor linking that issue to other policy objectives, such as wall funding. “We need to do DACA clean,” she
said.  On Wednesday, Trump aides will meet with congressional leaders to discuss those issues. That will be followed by a weeke
nd of strategy sessions for Trump and Republican leaders on Jan. 6 and 7, the White House said. Trump was also scheduled to mee
t on Sunday with Florida Republican Governor Rick Scott, who wants more emergency aid. The House has passed an $81 billion aid
package after hurricanes in Florida, Texas and Puerto Rico, and wildfires in California. The package far exceeded the $44 billi
on requested by the Trump administration. The Senate has not yet voted on the aid.


LR Prediction: Not a Fake News
DT Prediction: Not a Fake News
GB Prediction: Not a Fake News
RF Prediction: Not a Fake News
```

In [1]:
```
import matplotlib.pyplot as plt

models = ['Logistic Regression', 'Decision Tree', 'Gradient Boosting', 'Random Forest']
accuracy = [0.9849, 0.9950, 0.9958, 0.9852]

plt.figure(figsize=(10,6))
plt.bar(models, accuracy, color=['blue', 'green', 'orange', 'red'])
plt.title('Model Accuracy Comparison')
plt.xlabel('Models')
plt.ylabel('Accuracy')
plt.ylim(0.98, 1.0)
plt.show()
```



In [32]:
```
from sklearn.metrics import confusion_matrix
import seaborn as sns

cm = confusion_matrix(y_test, pred_lr)
plt.figure(figsize=(8,6))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues',
            xticklabels=['Fake', 'True'],
            yticklabels=['Fake', 'True'])
plt.title('Confusion Matrix - Logistic Regression')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.show()
```



app.py 5

app.py > ...

```python
import streamlit as st
import pandas as pd
import re
import string
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.feature_extraction.text import TfidfVectorizer

# ------------------------------
# Preprocessing Function
def clean_text(text):
    text = text.lower()
    text = re.sub(r'\[.*?\]', '', text)
    text = re.sub(r'\\W', ' ', text)
    text = re.sub(r'https?://\S+|www\.\S+', '', text)
    text = re.sub(r'<.*?>+', '', text)
    text = re.sub(r'[%s]' % re.escape(string.punctuation), '', text)
    text = re.sub(r'\n', '', text)
    text = re.sub(r'\w*\d\w*', '', text)
    return text

# ------------------------------
# Load and prepare the data
@st.cache_data
def load_model():
    fake = pd.read_csv("Fake.csv")
    true = pd.read_csv("True.csv")
    fake["class"] = 0
    true["class"] = 1

    data = pd.concat([fake, true], axis=0)
    data = data.sample(frac=1).reset_index(drop=True)
    data = data.drop(['title', 'subject', 'date'], axis=1)

    data["text"] = data["text"].apply(clean_text)

    X = data["text"]
```

```
    y = data["class"]

    vectorizer = TfidfVectorizer()
    X_vec = vectorizer.fit_transform(X)

    model = LogisticRegression()
    model.fit(X_vec, y)

    return model, vectorizer

# ------------------------------
# Load Model
model, vectorizer = load_model()

# ------------------------------
# Streamlit UI
st.set_page_config(page_title="📰 Fake News Detector")
st.title("🧠 Fake News Detector")
st.markdown("Enter any  **News Article** below to check whether it's **REAL** or **FAKE**.")

user_input = st.text_area("📝 Paste News Text Here:", height=250)

if st.button("🔍 Check"):
    if user_input.strip() == "":
        st.warning("⚠️ Please enter some news content.")
    else:
        processed = clean_text(user_input)
        vector_input = vectorizer.transform([processed])
        prediction = model.predict(vector_input)[0]

        if prediction == 1:
            st.success("✅ This news is **REAL**.")
            st.balloons()
        else:
            st.error("🚫 This news is **FAKE**.")
            st.snow()
```

11. REFERENCES

(APA Format):

- Simplilearn. (n.d.). *Machine Learning with Python*. Retrieved from https://www.simplilearn.com
- Alex The Analyst. (n.d.). *YouTube Channel - Machine Learning and Data Science Tutorials*. Retrieved from https://www.youtube.com/@AlexTheAnalyst