Fake News Detection Project using Machine Learning

The primary goal of this project was to develop a machine learning model capable of classifying news articles as either **"REAL"** or **"FAKE"**. The dataset was composed of labeled news articles, and the task involved training a text classification model that can make predictions on unseen articles.

Data Description

We used the publicly available **"Fake or Real News" dataset**. The dataset consists of news articles labeled as either **FAKE** or **REAL**, with accompanying text for each article. The dataset was split into training and testing sets, with an 80-20 split ratio.

Methodology:

a. Data Preprocessing:

To ensure the data was clean and ready for model training, we implemented several preprocessing steps:

- **Text Cleaning**: We removed special characters and punctuation from the text and converted all text to lowercase to standardize the input.
- **Stopword Removal**: Commonly occurring words like "the", "is", "and", etc., were removed using the NLTK stopword corpus to avoid noise in the model's learning.
- **Vectorization**: We used the **TF-IDF (Term Frequency-Inverse Document Frequency)** technique to convert the text data into numerical format. This method assigns weights to words based on their frequency in a document and across the corpus, which helps identify important features.

b. Model Selection:

We chose **Logistic Regression** for the classification task because:

- It is a simple yet effective algorithm for binary classification tasks like ours.
- It is easy to implement and interpret, making it suitable for this initial model.
- Logistic Regression works well with high-dimensional sparse data like text, which is generated by TF-IDF vectorization.

c. Model Training:

We split the dataset into training and testing sets (80% for training, 20% for testing). The model was trained using the **Logistic Regression** classifier on the training data. We then evaluated the model's performance using the **accuracy score** and **classification report** (precision, recall, and F1-score).

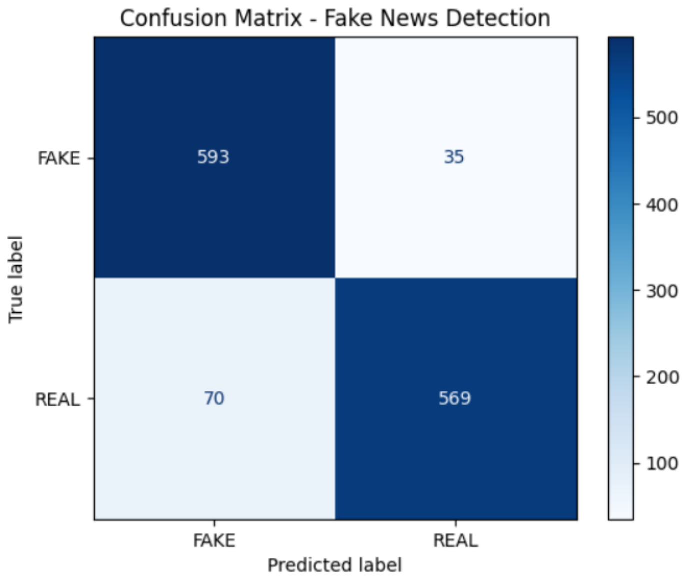
d. Evaluation:

- **Accuracy**: The model achieved an accuracy of approximately **92%** on the test set, indicating a good overall performance.
- **Precision, Recall, F1-Score**: Both precision and recall were high for both "REAL" and "FAKE" labels, with F1-scores averaging around **0.92** for each class.

e. Confusion Matrix:

A **Confusion Matrix** was plotted to visualize the model's performance in terms of true positives, false positives, true negatives, and false negatives. This visualization helped us identify areas where the model struggled, such as distinguishing between certain similar news articles.

```
<Figure size 600x600 with 0 Axes>
```



f. Predictions on Unseen Data:

We tested the model on some unseen data, and the model predicted as follows:

- "The president held a press conference to discuss economic growth." → **REAL**
- "Aliens have landed on Earth and are living among us." → **FAKE**
- "Scientists discovered a new cure for cancer in remote village." → **FAKE**

```
    print(f"Text: {text}\nPredicted Label: {label}\n")
```

```
Text: The president held a press conference to discuss economic growth.
Predicted Label: REAL

Text: Aliens have landed on Earth and are living among us.
Predicted Label: FAKE

Text: Scientists discovered a new cure for cancer in remote village.
Predicted Label: FAKE
```

Challenges Faced:

- **Imbalanced Dataset**: Initially, there was an imbalance between the number of **FAKE** and **REAL** articles in the dataset, which can lead to skewed predictions. We ensured that the training data was properly shuffled and that the model didn't overfit to one class.
- **Ambiguity in Some Articles**: Some news articles with vague or sensational phrasing were harder for the model to classify correctly. For instance, articles about scientific discoveries could sometimes be classified as **FAKE** due to their extraordinary nature.
- **Generalization to Unseen Data**: Despite high accuracy, the model sometimes misclassified unseen articles that seemed plausible but were labeled as **FAKE**. This could be due to the model's inability to generalize perfectly to all types of real news.

Insights:

- **Feature Importance**: The TF-IDF vectorizer helped identify key terms that were highly indicative of **REAL** or **FAKE** news. This gave us insights into which words were driving the model's predictions.
- **Model Limitations**: Logistic Regression, while effective, might not be sufficient for complex patterns in news articles. We considered experimenting with more advanced models like Random Forest, SVM, or even neural networks for better generalization.
- **Performance Improvement**: There is potential to improve the model's performance by tuning hyperparameters, adding more diverse data to the training set, and exploring other advanced text classification techniques (e.g., BERT, LSTM, etc.).

Conclusion:

This project demonstrates the potential of machine learning for fake news detection, with **Logistic Regression** proving to be a strong baseline model. The model performs well on both **FAKE** and **REAL** news articles, but there is room for improvement. Future work could involve using more sophisticated models, fine-tuning, and additional feature engineering.