Project Title: News Article Classification (Fake/Real)

Author: Mohd Amzad, M. Tech (Computer Engg)
Institution: Elevate Labs
Project Type: Internship Project

Introduction

In the era of digital journalism, misinformation spreads quickly and influences public discourse. This project tackles the issue by building a Fake News Detection system using Natural Language Processing (NLP) and supervised machine learning. The goal is to classify news articles as “Fake” or “Real” by identifying linguistic patterns and semantic inconsistencies often invisible to the human eye.

Python serves as the base language, supported by powerful libraries such as Scikit-learn (Sklearn), Pandas, and NLTK. The system processes raw news text through tokenization, stopword removal, stemming, and vectorization. These steps prepare the data for training models like Logistic Regression, Naive Bayes, and SVM. A user-friendly interface built with Streamlit allows real-time prediction, promoting public engagement and media awareness.

Tools Used

Each tool plays a unique role in the project pipeline:

- **Python:** Core development language enabling rapid iteration and seamless integration.
- **Pandas:** For efficient data loading, cleaning, and formatting.
- **Sklearn:** Hosts classification algorithms, feature selection, and model evaluation tools.
- **NLTK:** Provides essential NLP utilities like tokenization, lemmatization, and stopword filtering.
- **CountVectorizer / TfidfVectorizer:** Convert processed text into numerical vectors for modeling.
- **Streamlit:** Enables deployment of a user-facing interface for interactive classification.

Methodology Steps

A structured approach was followed:

1. **Data Collection:** Imported labeled CSV files from Kaggle with “Fake” and “Real” labels.
2. **Cleaning:** Removed duplicates, special characters, and missing entries using Pandas and regex.
3. **Preprocessing:** Performed tokenization, stopword removal, and stemming with NLTK.
4. **Feature Extraction:** Converted text into numeric vectors using CountVectorizer and TfidfVectorizer.

5. **Model Building:** Trained models like Logistic Regression, Naive Bayes, and SVM on processed data.
6. **Evaluation:** Analyzed accuracy, precision, recall, F1-score, and confusion matrix.
7. **Prediction:** Used trained models to classify unseen articles.
8. **Deployment:** Developed a Streamlit app that allows users to submit text and view real-time predictions.
9. **Optimization:** Iteratively improved model performance using GridSearchCV and explored additional algorithms.

Performance Evaluation

The trained models were assessed using common metrics:

- **Accuracy:** Measures overall correctness of predictions.
- **Precision and Recall:** Evaluate detection of fake vs. real articles.
- **F1 Score:** Balances precision and recall.
- **Confusion Matrix:** Gives detailed classification breakdown.

Results:

Model	Accuracy
Naive Bayes	93.71%
Logistic Regression	98.74%

These metrics confirmed the system's reliability in identifying fake news patterns across multiple articles.

Deployment Interface

Streamlit enhanced usability by making the classification tool interactive. Users can input article headlines or text, which the system processes through the same NLP pipeline used in training. Predictions are shown instantly, making the tool accessible to non-technical audiences and researchers alike.

Conclusion

This project demonstrates how NLP and machine learning can be effectively integrated to address the growing challenge of fake news. The system combines linguistic rigor with algorithmic intelligence to classify articles and promote media authenticity. Streamlit deployment further expands its utility as a public-facing educational resource. Future enhancements may include deeper models like LSTMs, expanded datasets across multiple languages, and integration with live news feeds for real-time detection.