Fake News Detection Report

Abstract

This report outlines the development of a machine learning-based detection of fake news, designed to classify news headlines into real or fake categories. Using Natural Language Processing (NLP) techniques and logistic regression, the application processes text data in search of patterns indicative of misinformation. The project covers a range of phases, namely data cleaning, model development, evaluation, and distribution via a Streamlit based web app, which allows both individual headline tests, as well as batch testing from CSV files. The model enjoys a significant degree of accuracy, demonstrating its potential value in practical applications as a countermeasure against misinformation.

Introduction

The spread of fake information presents significant challenges to the integrity of information, affecting public opinion and decision making. This project aims to develop a robust fake news detection system using machine learning and NLP techniques. Utilizing datasets with labeled fake and true news articles, the system applies a logistic regression model within a pipeline featuring TF-IDF vectorization for extracting text features. The completed model is launched as an interactive Streamlit app, allowing users to enter news headlines or upload CSV files for batch predictions. This report details the tools used, the steps involved and the results of the project.

Tools Used

The following tools and libraries were utilized in the development of the fake news detection:-

- **Python**: Core programming language for data processing, modeling, and application development.
- **Pandas and NumPy**: For data manipulation and numerical computations.
- **NLTK**: For text pre-processing, including stopword removal and tokenization.
- **Scikit-learn**: For TF-IDF vectorization, logistic regression modeling, and evaluation metrics.
- **Matplotlib and Seaborn**: For data visualization, including bar charts and heatmaps.
- **Joblib**: For saving and loading the trained model.
- **Streamlit**: For creating an interactive web application.
- **Pyngrok and Flask**: For hosting the Streamlit app with a public URL.

Steps Involved in Building the Project

The development process followed these steps:-

1. **Data Loading and Labeling**: Two datasets, *Fake.csv* and *True.csv*, were loaded using Pandas. Labels ('fake' or 'true') were added to each dataset to indicate the reliability of the article.
2. **Data Pre-processing**: The datasets were merged, and duplicates were removed. Unnecessary columns (e.g., date, title) were dropped. Text pre-processing included converting text to lowercase, removing punctuation, and eliminating stopwords using NLTK.
3. **Exploratory Data Analysis**: The distribution of articles by subject and target (fake vs. true) was visualized using bar charts. Word frequency analysis was conducted to identify common terms in fake and true news.
4. **Model Training**: A pipeline combining TF-IDF vectorization and logistic regression was created using Scikit-learn. The dataset was split into training (80%) and testing (20%) sets, and the model was trained on the training data.
5. **Model Evaluation**: The model achieved a training accuracy of approximately 99% and a test accuracy of around 98%. A classification report and confusion matrix were visualized using heatmaps to assess precision, recall, and F1-scores.
6. **Model Deployment**: The trained model was saved using Joblib. A Streamlit application was developed to allow users to input single headlines or upload CSV files for batch predictions. The app was hosted using Pyngrok for public access.

Conclusion

The fake news detection successfully classifies news articles with high accuracy because of its use of Natural Language Processing(NLP) and Logistic Regression to help distinguish between real and fake news articles as well as the pre-processing steps taken to ensure clean and relevant data for the dataset, and Streamlit provided interactive interface for users with the project. Further improvements to the project could be using advanced models such as transformers and possibly expanding the dataset to help with generalization. Overall, this project shows how machine learning can be used as a tool to addressing misinformation, as it offered a very relevant, user-friendly application to use for verifying news articles.