Project Completion Report: Fake News Prediction using Logistic Regression and NLP

October 27, 2025

Abstract

This report details the development of a predictive model for classifying news articles as 'Real' or 'Fake'. The project uses a dataset of 20,800 news articles, employing Natural Language Processing (NLP) techniques, specifically stemming and TF-IDF vectorization, to transform textual data into a machine-readable format. A Logistic Regression classifier was trained on this data. The final model achieved a high accuracy of **98.66**% on the training data and **97.91**% on the unseen test data, demonstrating robust performance in identifying deceptive information based primarily on the author and title of the articles.

1 Introduction

In the age of digital information, the rapid spread of misinformation poses significant challenges. This project addresses the problem of fake news detection by creating an automated system that classifies news articles. The approach leverages machine learning to identify patterns in the textual content (author and title) that correlate with the veracity of the news. This report documents the entire process, from data preprocessing using NLP techniques to model training and final evaluation.

2 Tools Used

The following key technologies and libraries were integral to this project:

- **Python**: The core programming language for implementation.
- **Pandas & NumPy**: Used for data loading, manipulation, and numerical processing of the dataset.
- **NLTK (Natural Language Toolkit)**: Utilized for text preprocessing, including the management of stopwords and the PorterStemmer for stemming.
- **Scikit-learn**: Provided essential machine learning components:
  - TfidfVectorizer for feature extraction (converting text to numerical data).
  - train_test_split for partitioning the dataset.
  - LogisticRegression for the primary classification model.
  - accuracy_score for model evaluation.
- **Regular Expressions (re)**: Used for cleaning text by removing non-alphabetic characters.

3 Steps Involved in Building the Project

The prediction system was developed through a structured NLP and machine learning pipeline:

Step 1: Data Preprocessing and Cleaning

1. **Data Loading**: The dataset, containing 20,800 news articles with columns for ‘id‘, ‘title‘, ‘author‘, ‘text‘, and ‘label‘ (0 for real, 1 for fake), was loaded into a Pandas DataFrame.
2. **Handling Missing Values**: Missing values in the ‘title‘, ‘author‘, and ‘text‘ columns were replaced with an empty string (”) to prevent errors during text processing.
3. **Feature Engineering**: The ‘author‘ name and ‘title‘ were merged to create a single, more informative feature called ‘content‘.
4. **Stemming (NLP)**: A custom function was applied to the ‘content‘ column for text cleaning:
    - Non-alphabetic characters were removed.
    - All text was converted to lowercase.
    - stopwords (common, low-value words like ’the’, ’a’, ’is’) were removed.
    - PorterStemmer was applied to reduce words to their root form (e.g., ’acting’ → ’act’).
5. **Data Separation**: The processed text data (**X**) and the numerical labels (**Y**) were separated into NumPy arrays.

Step 2: Feature Extraction and Model Training

1. **TF-IDF Vectorization**: The textual data (**X**) was converted into numerical feature vectors using TfidfVectorizer. This assigns a weight to each word based on its frequency in the document and its inverse frequency across all documents, reflecting the importance of the word.
2. **Train-Test Split**: The data was split into training and test sets using a $80 : 20$ ratio, with stratify $=$ **Y** to ensure an equal proportion of real and fake news in both sets.
3. **Model Training**: A LogisticRegression model was initialized and trained using the vectorized training data (**X_train**) and its corresponding labels (**Y_train**).

Step 3: Evaluation and Prediction

1. **Accuracy Measurement**: The model’s performance was evaluated using the accuracy_score:
    - Training Data Accuracy: **98**.**66**%
    - Test Data Accuracy: **97**.**91**%
2. **Predictive System**: The final code demonstrates the predictive capability by passing a new data point (**X_test**$[3]$) to the model. The model correctly predicted the news as ’Real’ (Label 0), matching the ground truth label.

4 Conclusion

The Fake News Prediction project successfully utilized NLP and a robust Logistic Regression model to classify news articles with high accuracy. The performance metrics, particularly the test accuracy of 97.91%, indicate that the model generalized well and effectively captured the linguistic patterns distinguishing real from fake news, primarily based on the combined author and title information. This model provides a strong baseline for automated fake news detection and could be further enhanced by incorporating the full article text and experimenting with deep learning architectures like LSTMs or Transformers.