Fake News Detection Using Logistic Regression on Campus News Dataset

Suraiya Mahmuda
Department of Computer Science and Engineering
Jahangirnagar University, Dhaka, Bangladesh

April 19, 2025

Abstract

Fake news has become a significant threat to the credibility of online information. This paper proposes a machine learning approach to detect fake news articles using Logistic Regression. A campus news dataset containing real and fake news is preprocessed and analyzed. Natural Language Processing (NLP) techniques like TF-IDF vectorization and stopwords removal are applied to extract meaningful features. The trained model achieves high accuracy in detecting fake news and demonstrates its potential to assist in automated fact-checking.

Keywords: Fake News Detection, Logistic Regression, Machine Learning, NLP, TF-IDF, Text Classification

1 Introduction

The proliferation of fake news, especially on social media and digital platforms, poses challenges for society, particularly in educational environments where misinformation can disrupt student and faculty perception. This study aims to implement a reliable and efficient fake news detection system using Logistic Regression to classify campus news as either real or fake.

2 Literature Review

Several machine learning algorithms like Naive Bayes, Decision Trees, and Deep Learning methods have been used for fake news detection. Logistic Regression, despite being a simpler model, provides fast and interpretable results, making it suitable for real-time detection scenarios. Prior works often focused on national or global news datasets; this paper focuses on a unique campus-specific dataset.

3 Methodology

3.1 Dataset

The dataset used includes campus-related news headlines, content, and a label indicating whether the news is "real" or "fake". It is stored in CSV format.

3.2 Data Preprocessing

Titles and content are merged into a single text field. Text is cleaned: converted to lowercase, URLs and special characters are removed. Stopwords are eliminated using the NLTK library.

3.3 Label Encoding

The target labels are encoded numerically:

- Real = 0
- Fake = 1

3.4 Feature Extraction

TF-IDF (Term Frequency–Inverse Document Frequency) is applied to convert the text into numerical vectors. A maximum of 5000 features is used to balance performance and accuracy.

3.5 Model Training

A Logistic Regression model is trained on 80% of the dataset, and tested on the remaining 20%. The model learns to associate patterns in the text with real or fake labels.

4 Implementation

The implementation is done in Python using libraries like pandas, scikit-learn, nltk, matplotlib, and seaborn. The main steps include:

- Cleaning the text data
- Removing stopwords
- Splitting the data into training and testing sets
- Applying TF-IDF vectorization
- Training a Logistic Regression model
- Evaluating with accuracy, confusion matrix, and classification report

A prediction function is also created to check new incoming news.

5 Results and Discussion

5.1 Evaluation Metrics

The model is evaluated using:

- Accuracy Score
- Confusion Matrix
- Precision, Recall, F1-Score from the classification report

5.2 Confusion Matrix

A heatmap visualization is used to display true positives, false positives, true negatives, and false negatives. This gives insight into model performance on unseen data.

5.3 Accuracy

The model achieved an accuracy of 100.00%, indicating it perfectly distinguishes real from fake campus news.

6 Conclusion

The Logistic Regression model successfully detects fake news in a campus context with high accuracy. Its simplicity and interpretability make it suitable for educational institutions looking to automate the verification of student news, notices, or announcements.

7 Future Work

Future enhancements can include:

- Using deep learning models (e.g., LSTM, BERT)
- Expanding the dataset to include multimedia elements
- Integrating the model into a real-time web or mobile app for live predictions

References

References

[1] A. Ahmed, "Fake News Detection on Social Media: A Data Mining Perspective," *ACM SIGKDD Explorations*, 2017.

[2] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, 1997.

[3] NLTK Documentation: `https://www.nltk.org`.

[4] Scikit-learn Documentation: `https://scikit-learn.org`.

[5] T. Joachims, "Text Categorization with Support Vector Machines," *ECML*, 1998.