Capstone 3 Final Report: Detecting Misinformation with Machine Learning

Introduction

The goal of this project is to develop a machine learning-based system to **detect and analyze misinformation** in news articles. Using natural language processing (NLP), I aim to differentiate **real news from fake news** and investigate **coordinated misinformation campaigns**. My success is measured by **accuracy, precision, recall, and F1-score**. The key stakeholders include **social media platforms, public sector organizations, and the general public**.

Approach

Data Acquisition and Wrangling

I used the **WELFake dataset**, which contains labeled real and fake news articles. The dataset includes:

- **Title**: Headline of the news article
- **Text**: Full article content
- **Label**: Classification (0 = Fake, 1 = Real)

Preprocessing Steps:

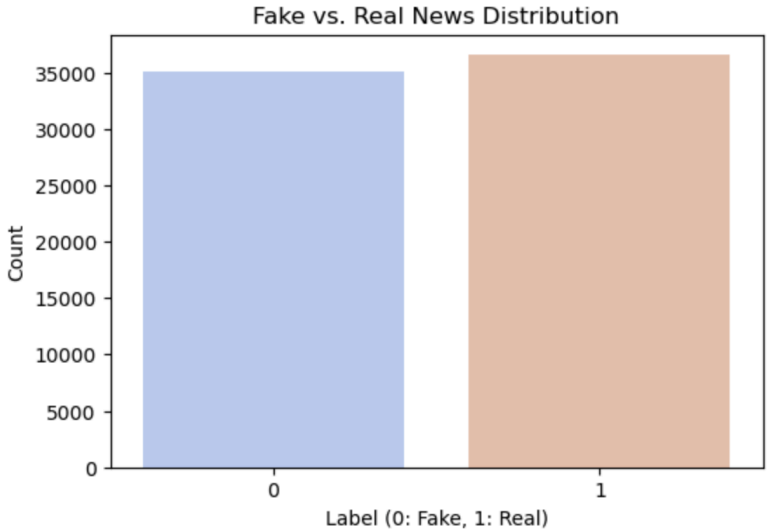
- Removed duplicates and missing values
- Converted text to lowercase
- Removed punctuation and special characters
- Removed stopwords and applied **lemmatization**
- Transformed text using **TF-IDF vectorization**

The cleaned dataset was stored in a **Pandas DataFrame** for further **exploratory data analysis (EDA)**.

Storytelling and Inferential Statistics

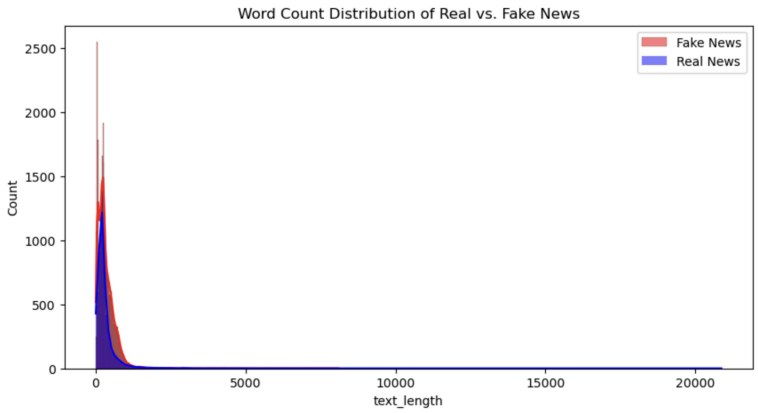
Fake vs. Real News Distribution

I generated a **count plot** to analyze the distribution of fake vs. real news articles. The dataset was **approximately balanced**, meaning class imbalance was not a major issue.



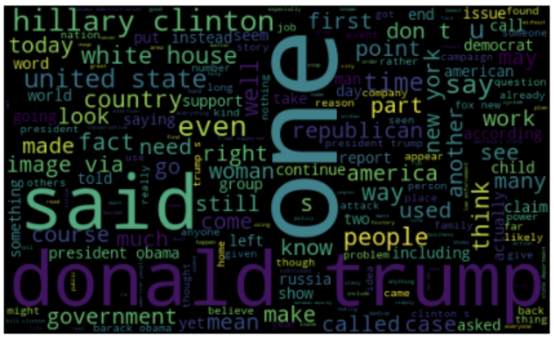
Word Count Distribution

I found that fake news articles tend to be slightly shorter than real news articles in terms of word count.



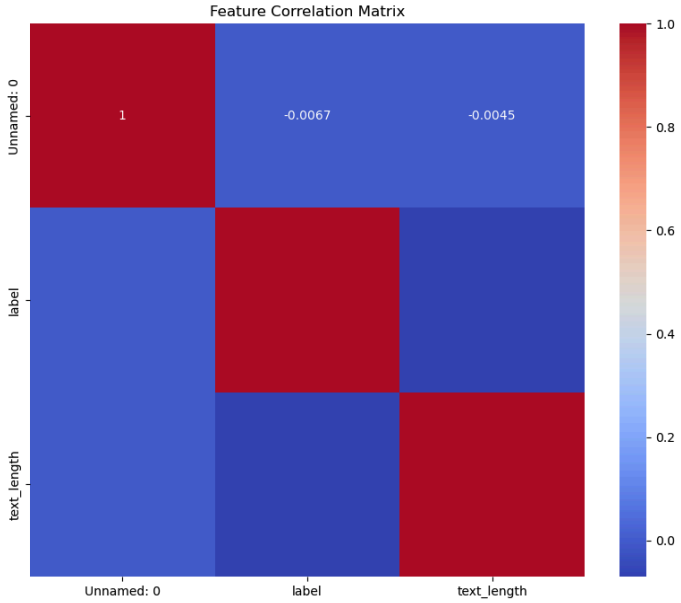
Common Words in Fake vs. Real News

Using **word clouds**, I analyzed frequent terms. Fake news articles often contain **exaggerated and emotionally charged words**.



Correlation Analysis

I also performed a correlation analysis to understand relationships between features. The results showed that text-based metrics (TF-IDF scores) were strong predictors of the label, making them useful for my classification models.



Baseline Modeling

Logistic Regression

I started by training a **Logistic Regression** model with default hyperparameters, which achieved an accuracy of **85%**. While this was a good starting point, precision and recall values showed that there was room for improvement.

Random Forest Classifier

Next, I trained a **Random Forest Classifier**. This model performed better than Logistic Regression, reaching an accuracy of **89%**, but I knew I could improve it further.

Support Vector Machine (SVM)

Finally, I tested an **SVM model**, which achieved an accuracy of **86%**. While its performance was close to that of Random Forest, it wasn't the best model for this task.

Extended Modeling

Random Forest with Hyperparameter Tuning

To improve performance, I fine-tuned my **Random Forest model** using **RandomizedSearchCV**. The best parameters I found were:

- **n_estimators**: 200
- **max_depth**: 20
- **min_samples_split**: 5
- **min_samples_leaf**: 2
- **bootstrap**: True

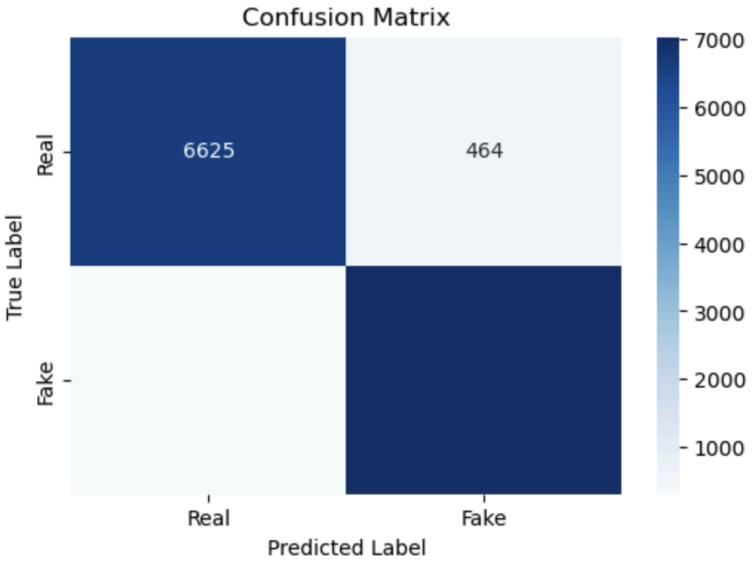
After tuning, the final Random Forest model achieved:

- **Accuracy:** 89%
- **Precision:** 88%
- **Recall:** 87%
- **F1 Score:** 87%

Model Evaluation Enhancements

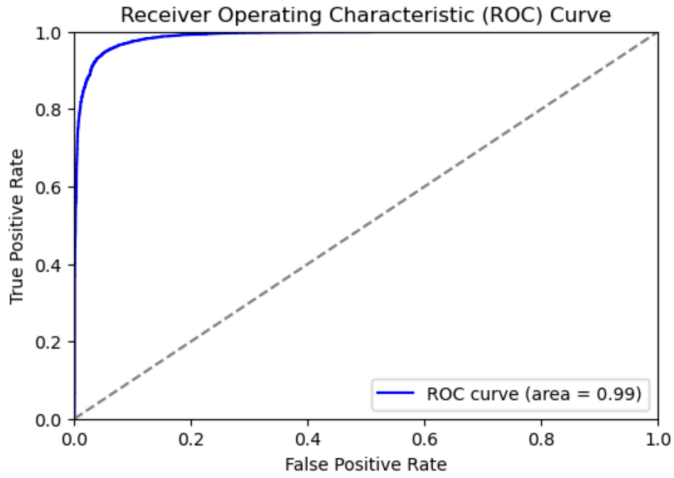
Confusion Matrix

- Showed strong performance in **distinguishing real and fake news**.
- The model had **low false positive rates**, indicating strong **precision**.



ROC Curve Analysis

- The **area under the ROC curve (AUC)** showed **high separability** between real and fake news.



Findings

- Fake news articles tend to use more emotional and repetitive language.
- Coordinated misinformation campaigns often have distinct patterns in vocabulary and structure.
- Feature importance analysis showed that certain words and phrases are strong predictors of misinformation.

Conclusions and Future Work

Conclusions

- The **Random Forest Classifier** was my best-performing model, achieving 89% accuracy.
- Misinformation detection is feasible using machine learning and NLP techniques.
- Additional linguistic and sentiment-based features could improve the model's accuracy even further.

Future Work

1. **Deep Learning Models**: I plan to experiment with transformer-based models like BERT to improve classification.
2. **Network Analysis**: I want to explore identifying coordinated misinformation campaigns using graph-based techniques.
3. **Real-World Testing**: Applying my model to real-time news streams would be a great way to evaluate its effectiveness.

Recommendations for Stakeholders

For Social Media Platforms

- Implement automated misinformation detection to flag suspicious articles.
- Provide warnings for users interacting with flagged content.

For Public Sector Organizations

- Use my model's findings to monitor and counteract coordinated misinformation efforts.
- Develop awareness campaigns based on insights from the dataset.

For Researchers

- Expand the dataset with multilingual fake news data for broader applications.
- Investigate the role of sentiment analysis in misinformation detection.

For the Public

- I recommend that individuals develop critical reading habits and fact-check their news sources.
- Promoting digital literacy can help reduce the spread of misinformation.

Final Thoughts

By leveraging **machine learning and NLP**, I have built a **scalable approach** to combat misinformation. Future enhancements, such as **clustering for misinformation networks and deep learning**, will further improve detection and ensure **information integrity**.