Fake news classifier:

Dataset used: Buzzfeed real and fake news dataset

. Introduction

Fake news has become a significant issue in today's media landscape, leading to the spread of misinformation. In this project, we aim to develop a machine learning model to classify news as real or fake, using the BuzzFeed dataset. By analyzing text features from the titles and bodies of news articles, we aim to identify patterns that differentiate real news from fake news. The dataset consists of 182 news articles across multiple sources.

Dataset Overview

The dataset contains 182 articles with 13 features each, covering aspects like the text of the article, the title, the author, and the source. The absence of any missing values (NA values), makes data cleaning minimal.

- The sources include well-known outlets such as *addictinginfo.org*, *rightwingnews*, and *freedomdaily*.
- We observed that some news sources, such as *rightwingnews*, predominantly contribute to fake news, while others, like *addictinginfo.org*, report more real news but with fewer articles.
- Interestingly, some articles are labeled as fake news even though their source is unknown, which might suggest that fake news can come from less-established or anonymous outlets, whereas real news tends to come from well-known sources.

Analysis of News Sources

- The data shows that **rightwingnews** reports the most fake news overall, but it also publishes some real news.
- **freedomdaily**, which is the second largest fake news reporting source, reports very little real news.
- **Addictinginfo.org** is the only source that reports more real news than fake news, although the total number of articles from this source is low.

There are **eight common sources** that report both real and fake news. For these, fake news articles dominate in volume compared to real news, with **rightwingnews** being the top fake news contributor.

Images and Movie Clips in Articles

- Most of the articles, whether real or fake, do not contain movie clips. However, articles without movie clips are more prevalent among fake news.
- **Images** appear in all real news articles, suggesting that real news tends to be better supported with visual proof. While fake news articles also include images, the correlation is not as strong.

This analysis suggests that images can be an important distinguishing factor between real and fake news, although movie clips provide less information.

Text Preprocessing

To prepare the text data for analysis, we applied several preprocessing steps:

- **clean_text Function**: This function uses `re.sub()` to replace special characters with spaces.
- **preprocess_text Function**:
    - Converts text to lowercase.
    - Removes numbers, punctuation, and additional special characters.
    - Tokenizes the text and removes stopwords.
    - Applies stemming using the **SnowballStemmer**.
    - Joins the cleaned tokens back into a single string.

These steps helped us clean the text data and reduce the noise in our model inputs.

Title Length Analysis

A statistically significant difference (p-value = 0.00225) was observed in the lengths of titles for real and fake news articles. On average, the titles of fake news articles are longer than those of real news:

- Mean title length for fake news: **8.33 words**
- Mean title length for real news: **7.24 words**

This suggests that fake news tends to use longer titles, potentially as a way to attract more attention.

N-grams and Sparse Terms

After processing the text by removing common stopwords and applying stemming, we analyzed **unigrams** (single words) and planned to analyze **bigrams** (word pairs). To retain the most meaningful terms while avoiding overfitting, we applied a **sparsity threshold** using `CountVectorizer` to remove infrequent terms. For example:

- A **max_df of 0.997** retains terms that appear in at least 0.3% of the documents.
- A **max_df of 0.97** retains only the top 24 terms that appear in at least 3% of the documents.

This helps us reduce the number of features, making our model more efficient and less prone to overfitting.

Classifier Models

We used three machine learning classifiers: **Logistic Regression**, **Naive Bayes**, and **Random Forest**. Each classifier was trained on three sets of features: titles only, bodies only, and a combination of both.

Logistic Regression

- **Why use it?** Logistic Regression is a simple yet effective classifier for binary classification problems. It is easy to interpret and works well when the relationship between the features and the target variable is linear.
- **Benefits**:
    - Interpretable: Logistic regression provides coefficients that give insights into how different terms contribute to the prediction.
    - Efficient: It is computationally lightweight, making it suitable for smaller datasets like ours.
    - Works well with high-dimensional data: Logistic Regression can handle sparse text data, especially after feature extraction (e.g., TF-IDF or count vectors).

Naive Bayes

- **Why use it?** Naive Bayes is a probabilistic classifier based on Bayes' Theorem. It assumes independence between features, which often holds well in text classification tasks despite the simplicity of the assumption.
- **Benefits**:
    - Robust for text classification: Naive Bayes often works well with high-dimensional data like text and is effective in distinguishing between fake and real news.
    - Fast and scalable: It has low computational complexity, making it ideal for working with large vocabularies in the dataset.
    - Handles small datasets well: Despite the small number of samples, Naive Bayes can still produce good results by leveraging conditional probabilities.

Random Forest

- **Why use it?** Random Forest is an ensemble method that combines multiple decision trees to improve prediction accuracy. It is particularly useful when the data contains non-linear relationships.
- **Benefits**:
    - Handles complex data: Random Forest can capture non-linear interactions between features, which is important when analyzing both the title and body of articles.
    - Reduces overfitting: By averaging multiple decision trees, Random Forest reduces the risk of overfitting, especially in small datasets.
    - Feature importance: Random Forest provides insights into which terms (features) are the most important for classifying real and fake news.

We built three classifiers off these models to analyze different aspects of the text in the news articles:

1. **Title-Based Classifier**
    - **Why use it?** Titles are the first thing readers see, and fake news articles often use exaggerated or misleading titles to grab attention. By analyzing the words and patterns in the titles, we can identify subtle cues that differentiate real news from fake news. For instance, fake news titles might be longer, contain more sensational language, or include certain keywords that are designed to elicit strong reactions.
    - **Purpose**: The goal of this classifier is to see if **title-based features** alone can predict whether an article is real or fake. Since titles are typically shorter than article bodies, the model has fewer terms to analyze, which makes it computationally less intensive. However, title information alone might not be sufficient for very accurate predictions.
    - **Benefits**:
        - Fast and lightweight: The title is a much shorter text compared to the full body, which makes feature extraction faster.
        - Effective for clickbait detection: Titles are often crafted to attract clicks, which can be a strong indicator of fake news.
        - Easier interpretation: The patterns found in the title can be more straightforward to analyze (e.g., sensational words or exaggerated phrases).
2. **Body-Based Classifier**
    - **Why use it?** While titles are important, the body of the article contains the full context and information being conveyed. Fake news articles often contain misleading information or disinformation that goes beyond just the title. By analyzing the text in the article body, we can capture the actual content of the news and detect inconsistencies, manipulative language, or specific terms associated with fake news.
    - **Purpose**: This classifier focuses on the **full text of the news article** to find patterns that predict whether the article is real or fake. The body of the article provides much more data, allowing for a richer feature set, but also introduces challenges like increased computational cost and complexity.
    - **Benefits**:
        - More comprehensive: The body of the article includes the full narrative, allowing for deeper analysis of content manipulation or bias.
        - Higher accuracy potential: Since it has more information to work with, the model may perform better at distinguishing real vs. fake news compared to just using the title.
        - Detection of nuanced patterns: Fake news often uses certain language patterns, emotional appeals, or fabricated details that might not be obvious in just the title.
3. **Combined Classifier**

- **Why use it?** Using both the title and the body together allows the model to leverage the benefits of both. Some fake news articles might have sensational or misleading titles but more subtle (or even seemingly legitimate) content in the body. On the other hand, some articles might use more balanced titles but contain fake information in the body. By combining these two sources of data, the model can make more informed predictions.
- **Purpose**: The combined classifier aims to **use both title and body features** to provide a more robust prediction. By integrating these two sources of information, the model can capture both the attention-grabbing elements of fake news (titles) and the deeper content-based manipulation (article body).
- **Benefits**:
    - More holistic analysis: By combining title and body features, the model gets a fuller picture of the article, leading to potentially more accurate predictions.
    - Cross-validation of signals: A strong indicator in the title (e.g., sensational wording) combined with a weaker indicator in the body might still point to fake news, and vice versa.
    - Improved robustness: By leveraging both short-text (title) and long-text (body) features, the model can be less sensitive to cases where one aspect (either title or body) alone would lead to a misclassification.

Why Use Multiple Models?

Using these three classifiers helps in several ways:

- **Comparison of Information Sources**: By separating the title and body classifiers, you can analyze whether one part of the article (title vs. body) is more predictive of fake news. This can reveal useful insights (e.g., fake news titles are more often sensational, or fake news bodies contain more inflammatory language).
- **Comprehensive Approach**: A combined classifier gives you a broader view of the article's content. While individual classifiers may miss subtle signals, combining title and body features often strengthens the model's performance.
- **Testing Hypotheses**: Using different classifiers allows you to test hypotheses, like whether fake news is more recognizable from titles or bodies, or whether using both improves overall accuracy.