Turing College

Petar Krešimir Borić

# Fake News Detection

2024/2025

# Intro

## Goal

**Fake News Detection** is a natural language processing task that involves identifying and classifying news articles or other types of text as real or fake.

The goal of fake news detection is to develop **algorithms** that can automatically identify and flag fake news articles, which can be used to combat misinformation and promote the dissemination of accurate information.

## About the dataset

Dataset separated in two files:

- Fake.csv (23502 fake news article)
- True.csv (21417 true news article)

Dataset columns:

- Title: title of news article
- Text: body text of news article
- Subject: subject of news article
- Date: publish date of news article

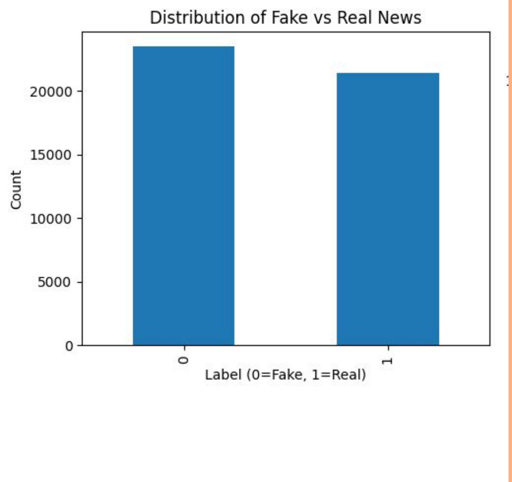
# EDA

## Class Distribution

**Total Articles:** 44,898

- **Fake News:** 23,481 (52.3%)
- **Real News:** 21,417 (47.7%)

**Observation:** Relatively balanced dataset, reducing class bias risks.



EDA

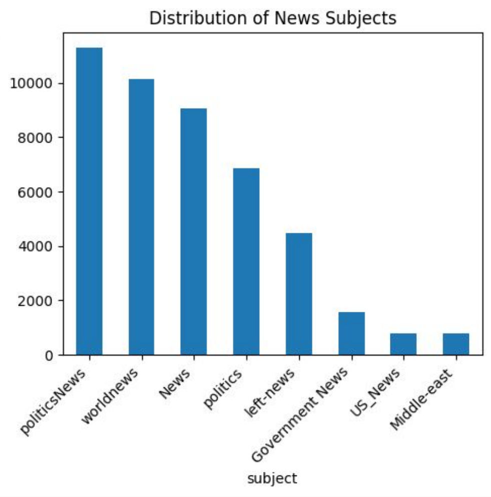
# Subject Analysis

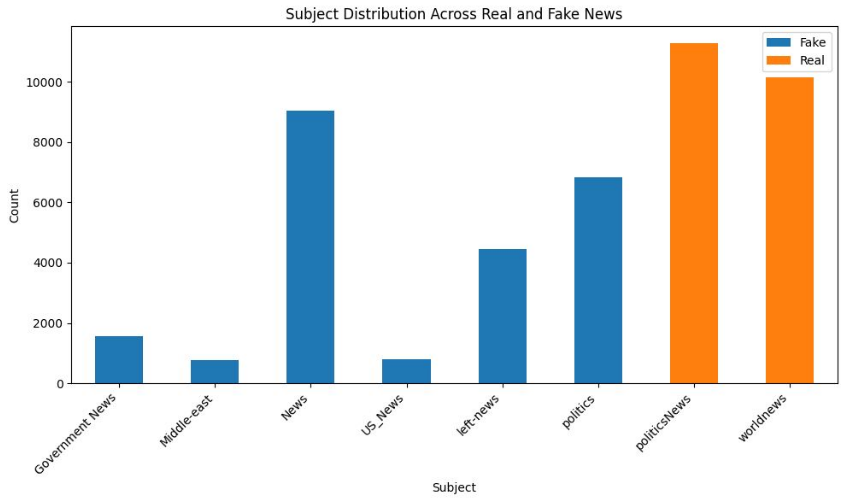
**Subjects Covered:** 8 Categories

1. Politics News (11,272)
2. World News (10,145)
3. News (9,050)
4. Politics (6,841)
5. Left News (4,459)
6. Government News (1,570)
7. US News (783)
8. Middle East (778)

**Key Observations:**

- **Dominance:** Political news is the most prevalent.
- **Imbalance:** Uneven distribution across subjects.
- **Correlation:** Some subjects exclusively contain real or fake news, indicating a strong correlation between subject and label.





EDA

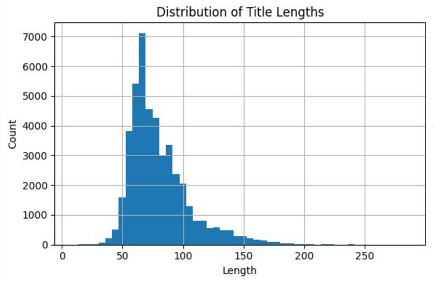
# Text Length Analysis

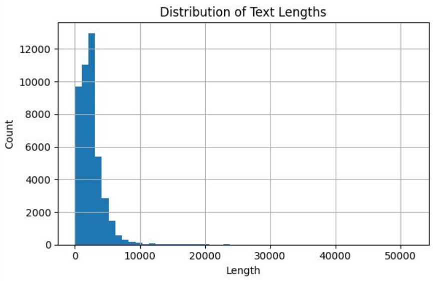
### Title Length

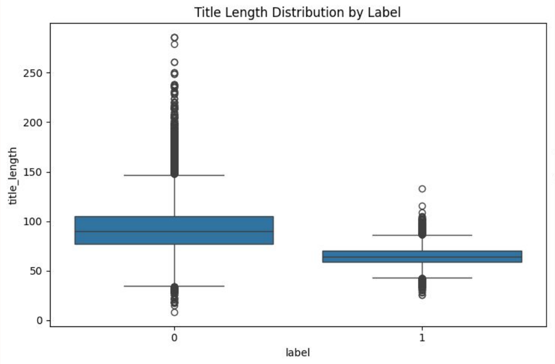
- **Findings:**
  - Fake news titles are generally longer (median ~85 characters) than real news titles (median ~60 characters).
  - Presence of extreme outliers in both categories.
- **Implication:** Title length could be a useful feature for classification.

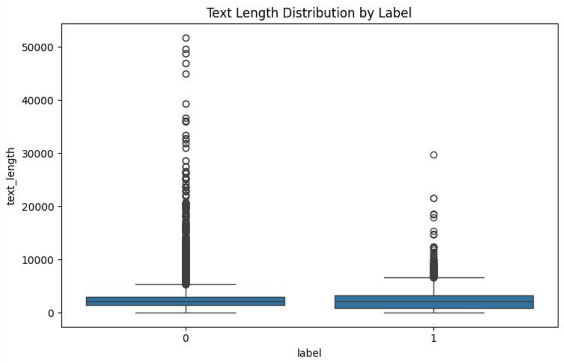
### Article Text Length

- **Findings:**
  - Similar distributions for both real and fake news.
  - Significant outliers present.
- **Implication:** Text length alone may not effectively distinguish between classes.


Distribution of Title Lengths


Distribution of Text Lengths


Title Length Distribution by Label


Text Length Distribution by Label

EDA

# Basic Statistics & Example Texts

**Statistics:**

- **Subject Distribution:** Overview of counts per subject.
- **Label Distribution:** Count of real vs. fake news.

**Example Articles:**

- **Real News Example:**
  - **Title:** As U.S. budget fight looms, Republicans flip their fiscal script
  - **Text:** WASHINGTON (Reuters) - The head of a conservative Republican faction in the U.S. Congress, who vote…
- **Fake News Example:**
  - **Title:** Donald Trump Sends Out Embarrassing New Year’s Eve Message; This is Disturbing
  - **Text:** Donald Trump just couldn t wish all Americans a Happy New Year and leave it at that. Instead, he had to give a shout out to his enemies, haters…

| Category | Count |
|---|---|
| **Subject Distribution** | |
| Politics News | 11,272 |
| World News | 10,145 |
| News | 9,050 |
| Politics | 6,841 |
| Left News | 4,459 |
| Government News | 1,570 |
| US News | 783 |
| Middle East | 778 |
| **Label Distribution** | |
| Fake News (0) | 23,481 |
| Real News (1) | 21,417 |

# Key Findings from EDA

**Dataset Balance:**

- Nearly balanced with a slight skew towards fake news.
- Advantageous for model training.

**Subject Correlation:**

- Strong correlation between subject and label.
- Potential for overfitting; requires careful handling.

**Text Characteristics:**

- Fake news tends to have longer, more sensational titles.
- Similar text lengths across classes.
- Distinct writing styles observed.

**Potential Features for Modeling:**

- Article subject
- Title length
- Text stylometric features
- Sentiment and emotional content
- Language patterns and complexity

# Model Development and Analysis

This section will cover the development of baseline models, traditional machine learning models, a BERT-based deep learning model, comprehensive model comparisons, error analysis, feature importance, and final conclusions and recommendations.

**Setup and Imports**

- **Libraries Utilized:**
  - **Data Manipulation:** `pandas`, `numpy`
  - **Visualization:** `matplotlib`, `seaborn`
  - **Text Processing:** `nltk`, `re`, `TextBlob`
  - **Machine Learning:** `scikit-learn`
  - **Deep Learning:** `PyTorch`, `PyTorch Lightning`, `Transformers`
- **Environment Configuration:**
  - Seed initialization for reproducibility
  - Downloading NLTK datasets

Model Development and Analysis

# Data Loading and Preprocessing

**Data Integration:**

- Combined `Fake.csv` and `True.csv` into a single DataFrame
- Assigned labels (`0` for fake, `1` for real)

**Text Cleaning:**

- Converted text to lowercase
- Removed non-alphabetic characters
- Tokenized and lemmatized words
- Eliminated stopwords

**Feature Engineering:**

- **Length Features:** Title and text lengths
- **Stylometric Features:**
  - Average word length
  - Punctuation count
  - Uppercase ratio
  - Sentiment polarity

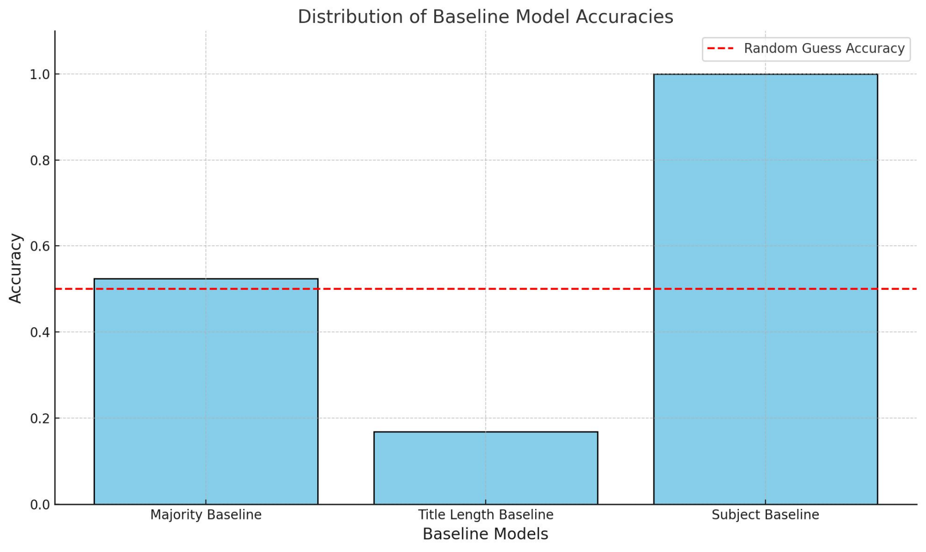
# Baseline Models

**Objective:**

- Establish performance benchmarks
- Serve as reference points for more complex models

**Implemented Baselines:**

1. **Majority Baseline:**
   - Always predicts the most common class
   - Introduces minor random variations to avoid complete bias
2. **Title Length Baseline:**
   - Uses the median title length to classify articles
   - Longer titles tend to indicate fake news
3. **Subject Baseline:**
   - Leverages the article's subject category for prediction
   - Certain subjects are more associated with fake or real news



# Traditional ML Models with Grid Search

**Models Explored:**

1. **Logistic Regression**
   - Regularization strength (`C`)
   - High `max_iter` for convergence
2. **Naive Bayes**
   - Smoothing parameter (`alpha`)
3. **Random Forest**
   - Number of trees (`n_estimators`)
   - Maximum depth (`max_depth`)

**Feature Engineering:**

- **Text Vectorization:**
  - **TF-IDF:** For Logistic Regression and Random Forest
  - **Count Vectorizer:** For Naive Bayes
- **Numerical Scaling:**
  - **StandardScaler:** For TF-IDF features
  - **MinMaxScaler:** For Count features

**Hyperparameter Tuning:**

- Employed `GridSearchCV` with 3-fold cross-validation
- Selected best parameters based on accuracy

Model Development and Analysis

# Model Training and Evaluation

**Training Process:**

- Trained each model with optimal hyperparameters from grid search
- Evaluated on an 80-20 train-test split

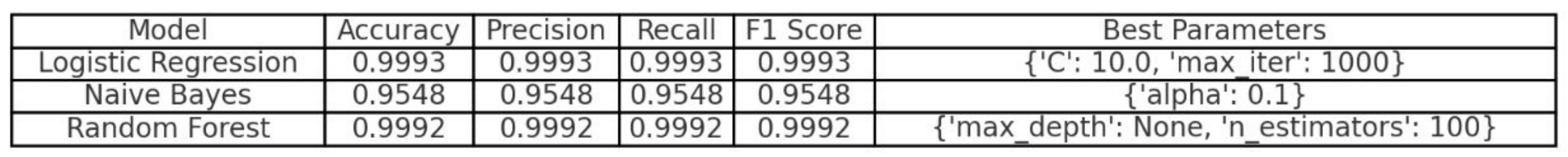
**Evaluation Metrics:**

- **Accuracy:** Overall correctness
- **Precision & Recall:** For Fake (`0`) and Real (`1`) classes
- **F1-Score:** Harmonic mean of precision and recall

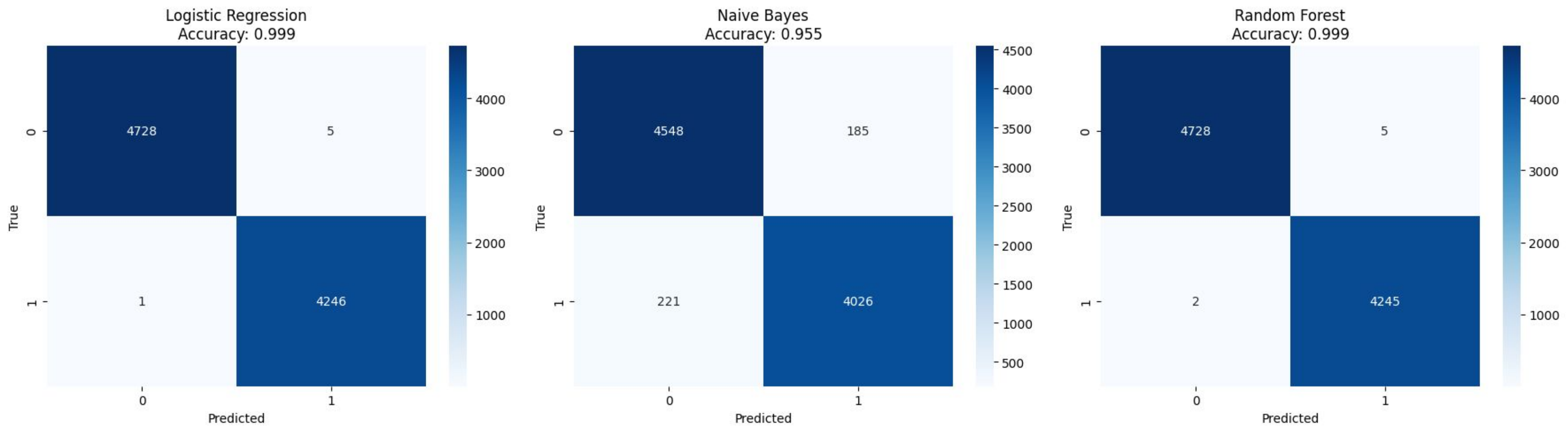
**Performance Overview:**

- All traditional ML models outperformed baseline models
- Logistic Regression and Random Forest achieved the highest accuracies

# Model Training and Evaluation - Performance Metrics



| Model | Accuracy | Precision | Recall | F1 Score | Best Parameters |
|---|---|---|---|---|---|
| Logistic Regression | 0.9993 | 0.9993 | 0.9993 | 0.9993 | {'C': 10.0, 'max_iter': 1000} |
| Naive Bayes | 0.9548 | 0.9548 | 0.9548 | 0.9548 | {'alpha': 0.1} |
| Random Forest | 0.9992 | 0.9992 | 0.9992 | 0.9992 | {'max_depth': None, 'n_estimators': 100} |



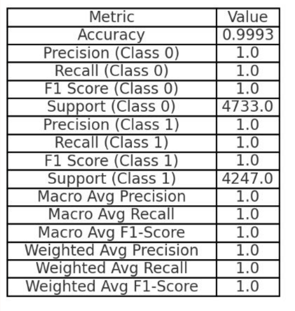
# BERT Model Training and Evaluation

### Training Highlights:

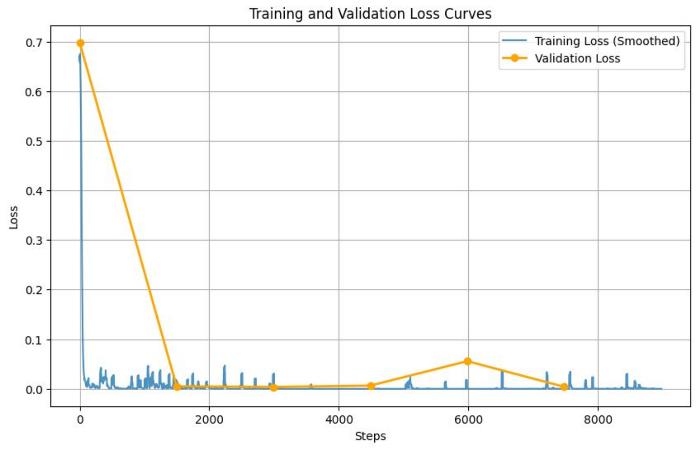
- Employed PyTorch Lightning for efficient training loops
- Implemented Early Stopping based on validation loss
- Trained for up to 5 epochs with potential early termination

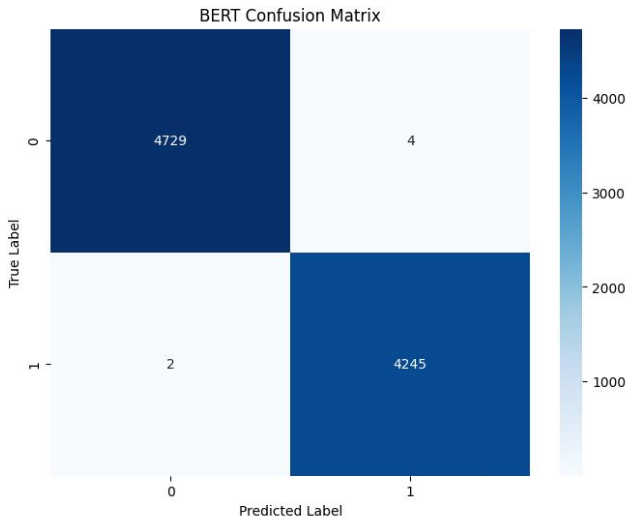
### Performance Metrics:

- **Accuracy:** Highest among all models
- **Precision & Recall:** Superior balance between classes
- **F1-Score:** Enhanced harmonic mean indicating robust performance



| Metric | Value |
|---|---|
| Accuracy | 0.9993 |
| Precision (Class 0) | 1.0 |
| Recall (Class 0) | 1.0 |
| F1 Score (Class 0) | 1.0 |
| Support (Class 0) | 4733.0 |
| Precision (Class 1) | 1.0 |
| Recall (Class 1) | 1.0 |
| F1 Score (Class 1) | 1.0 |
| Support (Class 1) | 4247.0 |
| Macro Avg Precision | 1.0 |
| Macro Avg Recall | 1.0 |
| Macro Avg F1-Score | 1.0 |
| Weighted Avg Precision | 1.0 |
| Weighted Avg Recall | 1.0 |
| Weighted Avg F1-Score | 1.0 |

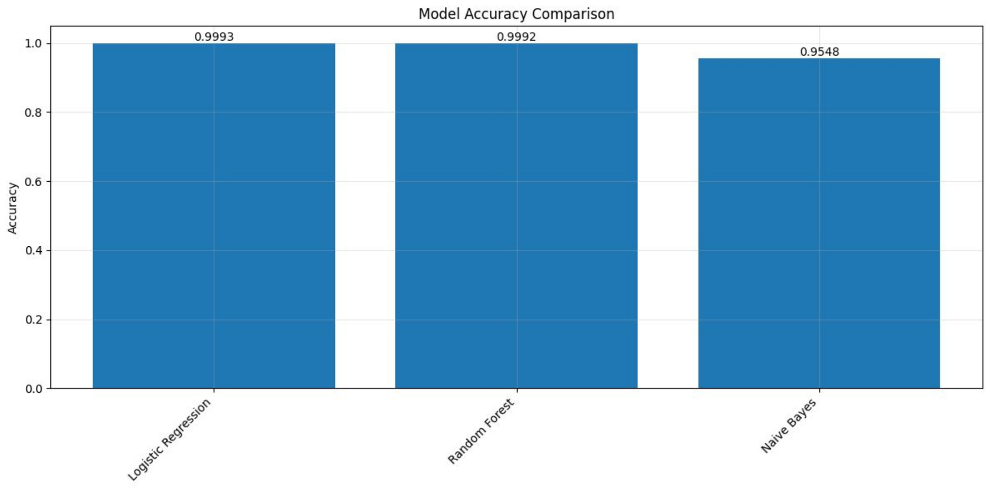


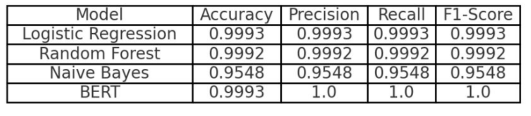


# Model Comparison and Analysis

**Insights:**

- **BERT** achieves the highest accuracy and F1-Score, demonstrating superior ability to understand contextual nuances.
- **Logistic Regression** and **Random Forest** offer strong performance with lower computational requirements.
- **Naive Bayes** provides competitive results with faster training times.
- Baseline models establish essential benchmarks, highlighting the effectiveness of advanced models.





| Model | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| Logistic Regression | 0.9993 | 0.9993 | 0.9993 | 0.9993 |
| Random Forest | 0.9992 | 0.9992 | 0.9992 | 0.9992 |
| Naive Bayes | 0.9548 | 0.9548 | 0.9548 | 0.9548 |
| BERT | 0.9993 | 1.0 | 1.0 | 1.0 |

Model Development and Analysis

# Error Analysis

**Objective:**

- Identify and understand the reasons behind misclassifications

**Findings:**

- **Neutral Language:** Articles lacking strong sentiment or biased language are often misclassified.
- **Mixed Signals:** Articles containing both credible and questionable information confuse models.
- **Similar Lengths:** Fake and real news articles with similar text lengths are harder to distinguish.

**Examples of Misclassifications:**

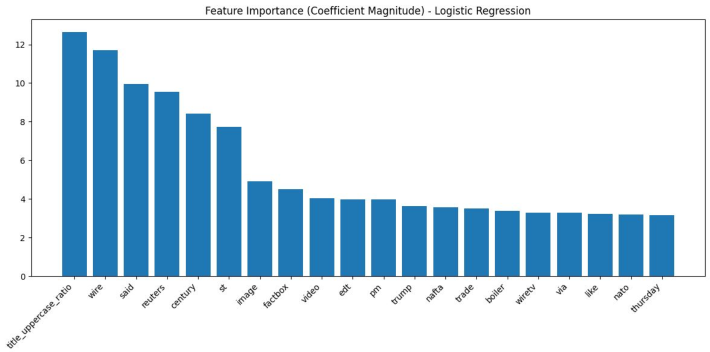
1. **Example 1:**
   - **True Label:** Real
   - **Predicted:** Fake
   - **Title:** generation gap china onechild generation grows
   - **Snippet:** "reuters class grew parent grandparent could dream food clothing comfort opportunity china economy surged childhood…"
2. **Example 2:**
   - **True Label:** Fake
   - **Predicted:** Real
   - **Title:** julian assange everything said he standing
   - **Snippet:** "st century wire say judging julian assange twitter page activity appears though may follow promise agree…"

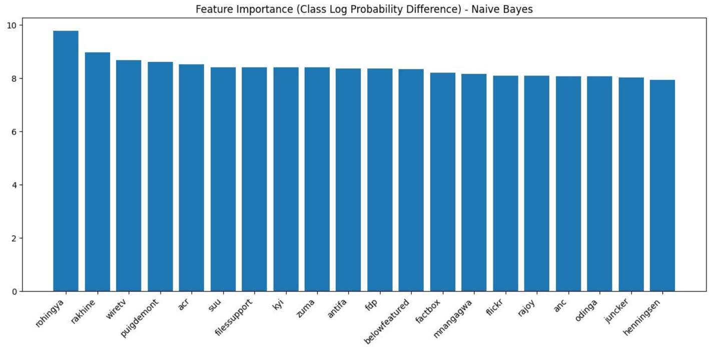
# Model Development and Analysis

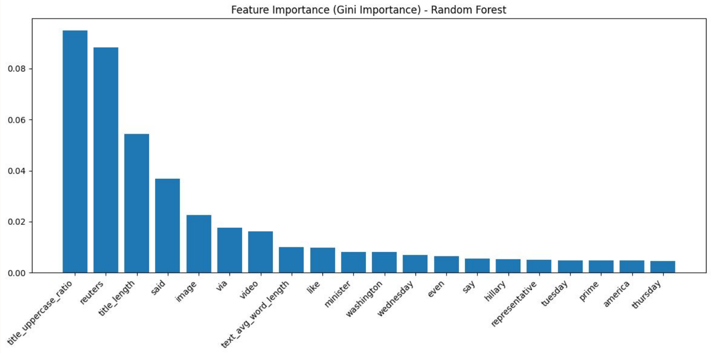
## Feature Importance Analysis

### Feature Importance Analysis

- **Key Insights:**
  - **Title Length:** Longer titles are more indicative of fake news across models.
  - **Top Keywords:** Words like "breaking," "exclusive," and "shocking" are strong indicators of fake news.
  - **Stylometric Features:** High punctuation counts and extreme sentiment scores are correlated with fake news.
- **Feature Importance by Model:**
  - **Logistic Regression:**
    - Coefficient magnitudes highlight the most influential features
  - **Random Forest:**
    - Gini Importance reveals non-linear feature contributions
  - **Naive Bayes:**
    - Probability differences between classes indicate discriminative power


Feature Importance (Coefficient Magnitude) - Logistic Regression


Feature Importance (Class Log Probability Difference) - Naive Bayes


Feature Importance (Gini Importance) - Random Forest

# Conclusions and Recommendations

**Model Performance Summary:**

- **BERT:**
  - **Pros:** Highest accuracy, excellent contextual understanding
  - **Cons:** High computational cost
- **Logistic Regression:**
  - **Pros:** Strong performance, low computational requirements
  - **Cons:** Limited ability to capture complex patterns
- **Random Forest:**
  - **Pros:** Good balance between performance and interpretability
  - **Cons:** May overfit with too many trees

**Key Findings:**

- **Effective Features:**
  - Title length, specific keywords, and stylometric features are critical
- **Challenges:**
  - Possible overfitting issue

**Recommendations:**

1. **Model Selection:**
   - **Deploy Logistic Regression** for efficient, scalable applications
   - **Utilize BERT** when the highest accuracy is essential and resources permit
2. **Enhance Feature Engineering:**
   - Integrate additional stylometric features like syntactic patterns
   - Incorporate source credibility and historical reliability metrics
3. **Future Improvements:**
   - **Dataset Expansion:** Gather more diverse and recent news articles
   - **Ensemble Methods:** Combine multiple models to leverage their strengths
   - **Temporal Validation:** Ensure models remain effective over time by validating on data from different periods