Fake News Detection

MAIM NLP Track Final Project

Presented by: Seif Eldeen Khalid Nabil

Course: MAIM NLP Track

Date: 9/4/2025

Abstract

This project, part of the **MAIM NLP Track Final Project**, presents a complete pipeline for fake news detection and sentiment analysis. The work is organized into **three Python notebooks**: the first covers classical machine learning models (Naïve Bayes, Logistic Regression, SVM), the second implements pretrained embeddings and sequential models (RNN, GRU, LSTM) with comparisons between **Word2Vec, GloVe, and FastText**, and the third focuses on Transformer-based models (BERT) and fine-tuning.

All notebooks include **exploratory data analysis, text preprocessing, and evaluation** using metrics such as accuracy, precision, recall, and F1-score. A **Flask deployment file** demonstrates the final pipeline, enabling real-time classification of news headlines or tweets. Ethical considerations, transparency, and responsible AI use are emphasized throughout the submitted work.

Introduction

In today’s digital era, misinformation and fake news have become a significant business and societal challenge. Social media platforms such as Twitter and Reddit allow content to spread rapidly, often without verification, leading to reputational risks, poor business decisions, and financial losses for companies, media outlets, and organizations. For instance, misleading news can influence stock prices, consumer behavior, or brand perception, resulting in wasted resources and operational inefficiencies.

The need for timely and accurate detection of fake news has never been greater. Traditional manual fact-checking methods are slow, costly, and cannot scale to the enormous volume of online content. Businesses spend substantial resources verifying information, monitoring social media, and mitigating the effects of misinformation.

This project addresses this problem by providing an **automated solution for detecting fake news and analyzing its sentiment impact**. By implementing a full Natural Language Processing (NLP) pipeline, organizations can quickly and reliably classify news articles or social media posts as real or fake. This solution reduces the dependency on manual labor, cuts operational costs, and allows resources to be reallocated to higher-value tasks.

Moreover, the automated system enables faster response times to misinformation, protecting brand reputation, maintaining public trust, and preventing financial losses caused by false information. By leveraging modern AI techniques, this project transforms the challenge of misinformation into a manageable business risk, offering measurable efficiency gains and cost savings for organizations in various sectors.

Dataset description:

The dataset for this project is a Fake News Detection dataset consisting of two CSV files:

- Fake.csv: Contains 23,502 fake news articles
- True.csv: Contains 21,417 real news articles

Each file contains the following columns:

- Title: The headline of the news article.
- Text: The main body of the article.
- Subject: The category or topic of the article (e.g., politics, world news).
- Date: The publish date of the article.

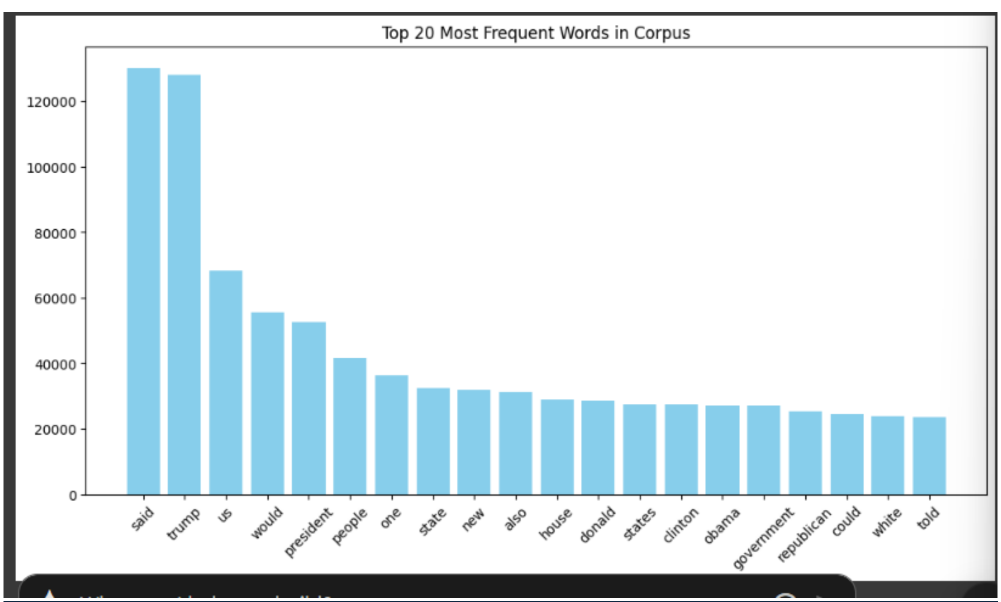
Data Processing:
To prepare the dataset for analysis and modeling:

1. A label column was added to each dataset (0 for fake news, 1 for real news).
2. Both datasets were concatenated into a single DataFrame, creating a combined dataset of 44,919 articles.
3. This combined dataset was then cleaned and preprocessed for analysis and modeling.

Exploratory Data Analysis (EDA) and Visualizations

1. Word Clouds:

- Word clouds were generated for both real and fake news articles to visualize the most frequent words in each category.
- In real news, common words reflect topics like "politics," "government," and "world events."
- In fake news, frequent words include sensational, politically biased, or attention-grabbing terms.



2. Subject Distribution:

- Real news:

  Subject | Count

  politicsNews : 11,272

  worldnews :10,145

- Fake news:

  Subject | Count

  News : 9,050

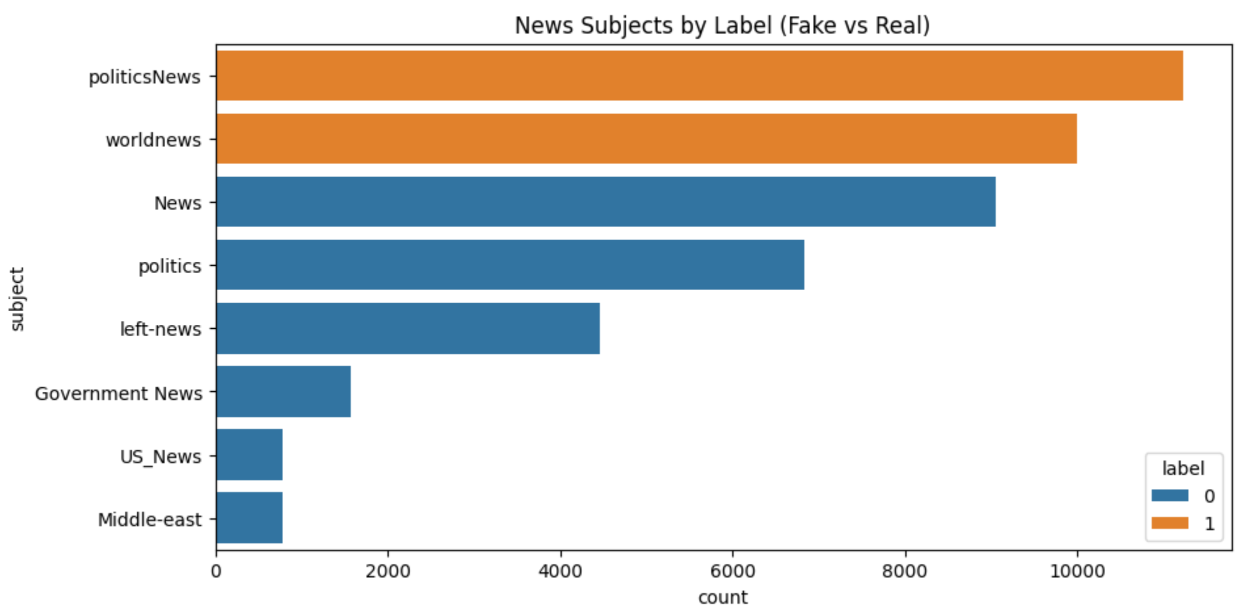
  politics : 6,841

  left-news : 4,459

  Government News: 1,570

  US_News : 783

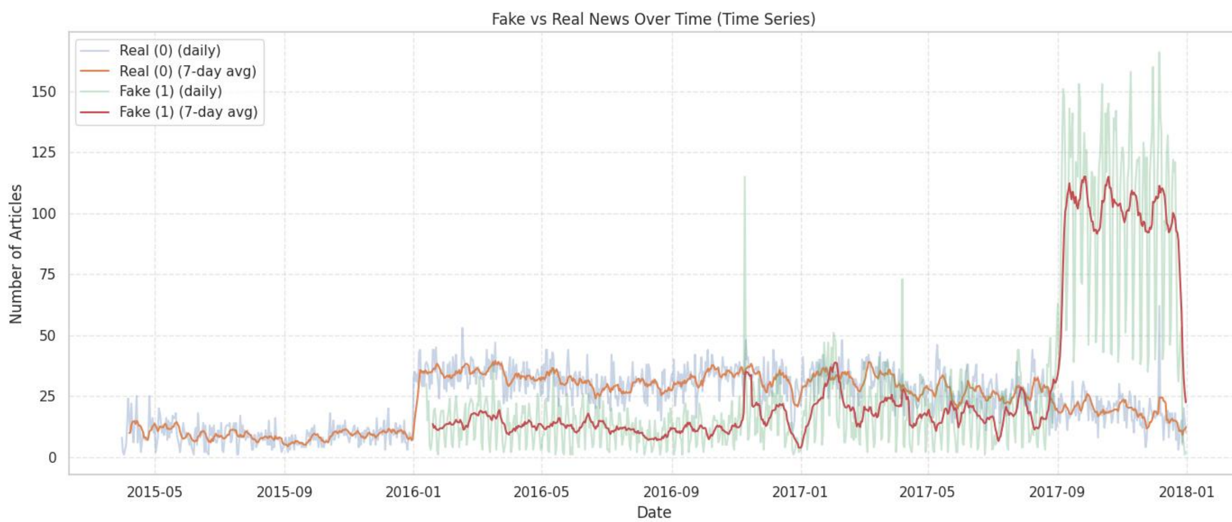
  Middle-east : 778



3. Fake vs Real News Over Time:

- A time series plot shows the number of fake vs real news articles over time.

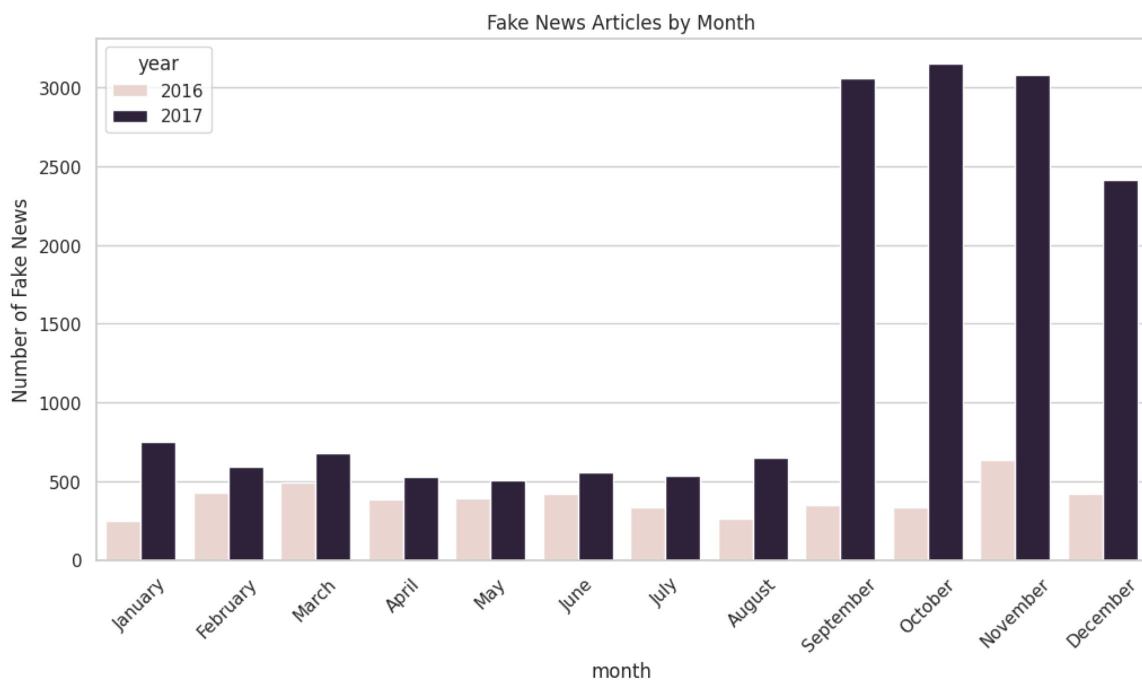
- There is a notable increase in fake news during 2017, particularly in September, October, November, and December.



4. A Bar plot shows the number of

fake vs real news articles over time.

There is a notable increase in fake news during 2017, particularly in September, October, Novembe and December.



Business Relevance:

- Analyzing subjects, frequency, and temporal patterns allows organizations to allocate monitoring resources efficiently.
- Word cloud and subject distribution visualizations reveal recurring themes in fake news, which can inform automated detection and human verification strategies.
- Temporal analysis supports strategic planning to reduce financial and reputational risks caused by misinformation.

Methodology

This project follows a structured **end-to-end approach** for detecting fake news, from data preprocessing to model deployment. The methodology is divided into several stages, progressing from classical machine learning models to modern Transformer-based architecture, with a focus on maximizing accuracy and business value.

1. Data Preprocessing

Prior to modeling, all text data (both article body and titles) underwent thorough preprocessing to ensure high-quality input for machine learning and deep learning models. The following steps were applied:

1. Lowercasing: All text was converted to lowercase to standardize tokens and reduce duplication.
2. Noise Removal: URLs, email addresses, and non-alphabetic characters were removed to eliminate irrelevant content.
3. Tokenization: Text was split into individual words (tokens) to facilitate further processing.
4. Stopwords Removal: Common English stopwords were removed to focus on meaningful words.
5. Lemmatization with POS Tagging: Words were reduced to their base or dictionary forms, considering their part-of-speech (noun, verb, adjective, adverb) to maintain semantic meaning.

The preprocessing was applied to both the article body (Text) and headlines (Title) to generate cleaned versions suitable for feature extraction and modeling. This process ensures that the models learn relevant linguistic patterns rather than noise, improving classification performance.

---

2. Feature Extraction

Text data is converted into numerical representations for model consumption:

- **Bag of Words (BoW):** Represents text as a frequency count of words in each article.
- **TF-IDF (Term Frequency–Inverse Document Frequency):** Captures word importance relative to the entire dataset, reducing the impact of common words.
- **Pretrained Word Embeddings:** Word2Vec, GloVe, and FastText embeddings were applied to sequential models to capture semantic meaning and context.
- **Comparing the performance of the Pretrained word embeddings**

```
Word2Vec Embedding Results:
Accuracy: 0.9243875278396436
              precision    recall  f1-score   support

           0       0.94      0.92      0.93      4644
           1       0.91      0.93      0.92      4336

    accuracy                           0.92      8980
   macro avg       0.92      0.92      0.92      8980
weighted avg       0.92      0.92      0.92      8980


FastText Embedding Results:
Accuracy: 0.9103563474387528
              precision    recall  f1-score   support

           0       0.92      0.90      0.91      4644
           1       0.90      0.92      0.91      4336

    accuracy                           0.91      8980
   macro avg       0.91      0.91      0.91      8980
weighted avg       0.91      0.91      0.91      8980


GloVe Embedding Results:
Accuracy: 0.9144766146993318
              precision    recall  f1-score   support

           0       0.92      0.91      0.92      4644
           1       0.91      0.92      0.91      4336
```

3. Classical Machine Learning Models

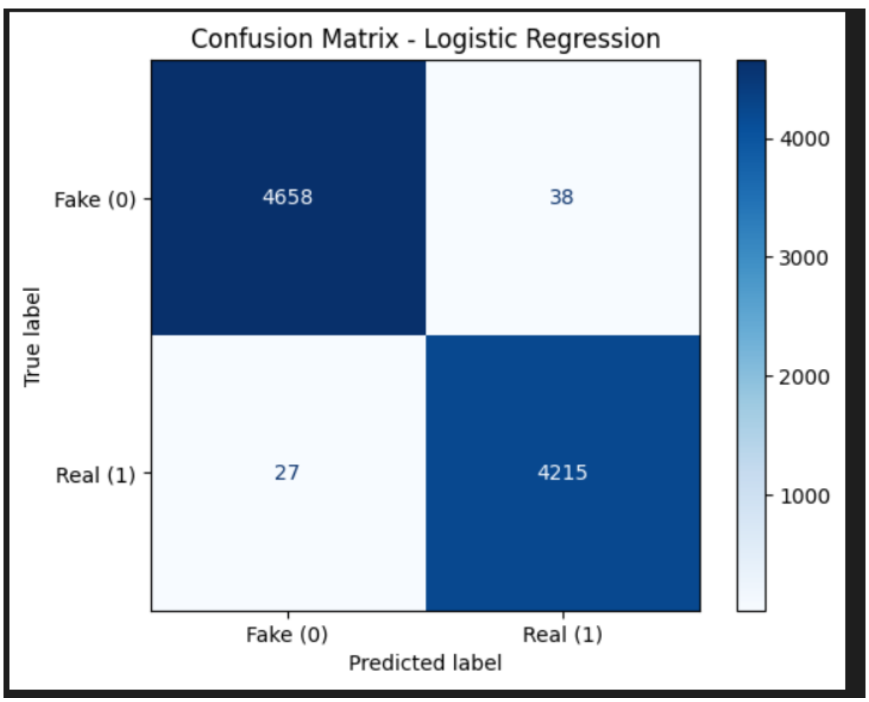
Baseline models were implemented to establish reference performance metrics:

- **Naïve Bayes:** Probabilistic classifier based on word frequencies.
- **Logistic Regression:** Predicts the probability of an article being real or fake using a linear model.
- **Support Vector Machines (SVM):** Separates classes in high-dimensional space for robust classification.

These models were trained using BoW and TF-IDF representations and evaluated using **accuracy, precision, recall, and F1-score**.

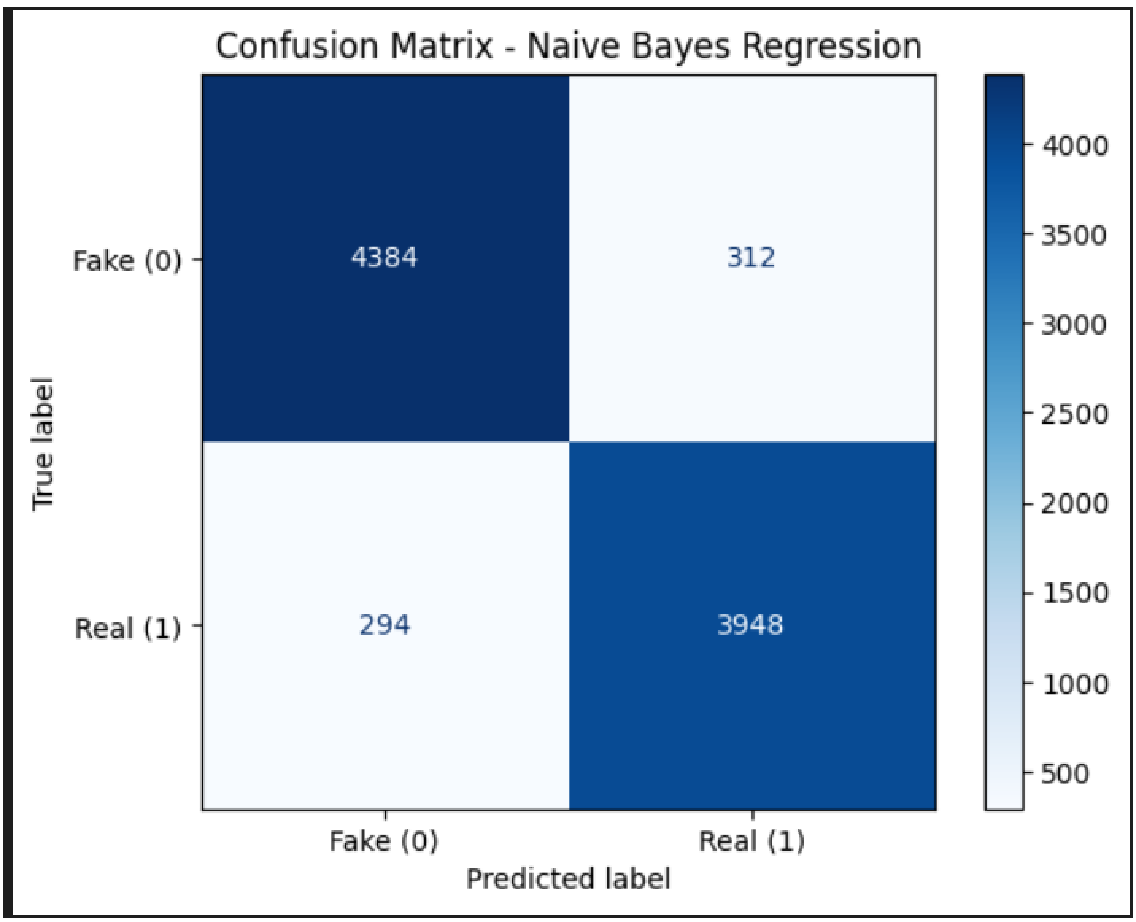
Logistic regression results:

```
Best parameters: {'clf__C': 10, 'tfidf__max_features': 10000}
Best F1-score: 0.9909705956180115
Test set performance:
              precision    recall  f1-score   support

           0       0.99      0.99      0.99      4696
           1       0.99      0.99      0.99      4242

    accuracy                           0.99      8938
   macro avg       0.99      0.99      0.99      8938
weighted avg       0.99      0.99      0.99      8938
```



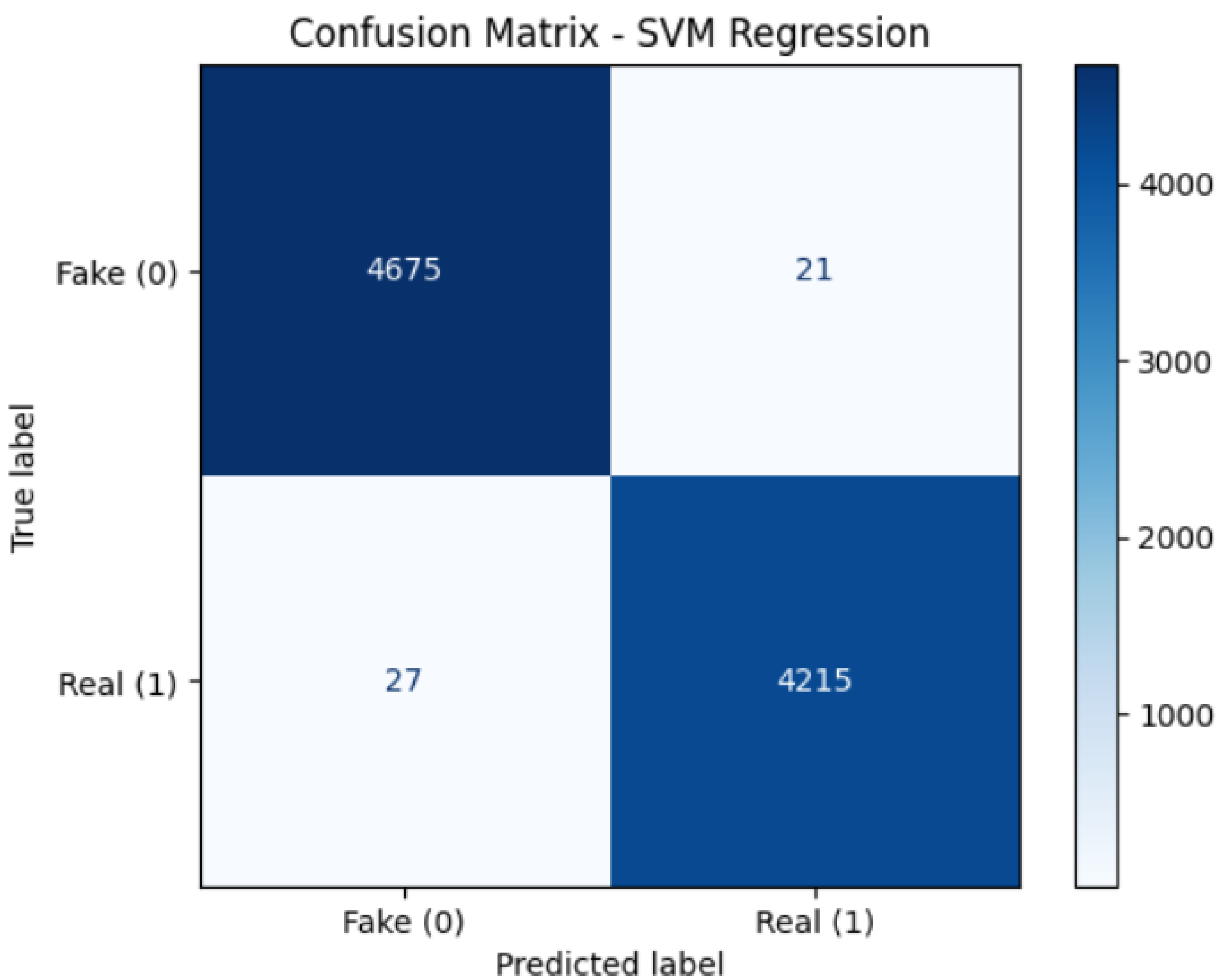
Naiive Bayes results:

```
Naive Bayes performance:
              precision    recall  f1-score   support

           0       0.94      0.93      0.94      4696
           1       0.93      0.93      0.93      4242

    accuracy                           0.93      8938
   macro avg       0.93      0.93      0.93      8938
weighted avg       0.93      0.93      0.93      8938
```



SVM results:

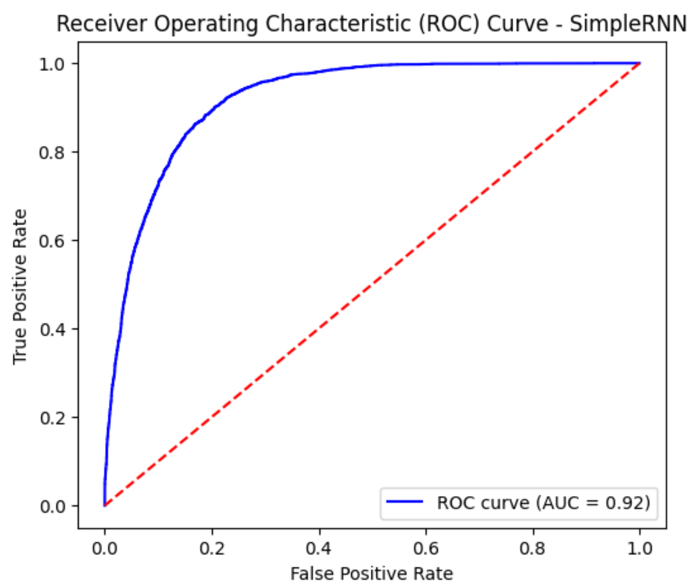
```
SVM performance:
              precision    recall  f1-score   support

           0       0.99      1.00      0.99      4696
           1       1.00      0.99      0.99      4242

    accuracy                           0.99      8938
   macro avg       0.99      0.99      0.99      8938
weighted avg       0.99      0.99      0.99      8938
```

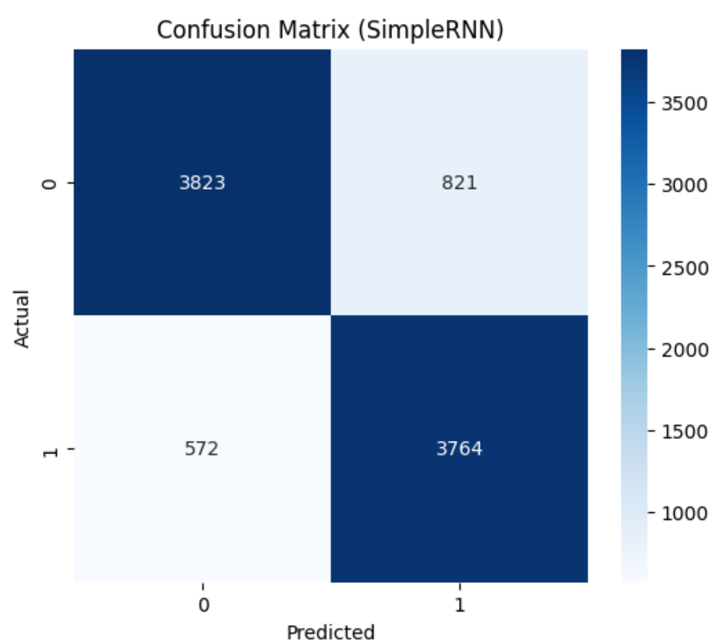


4. Sequential Deep Learning Models

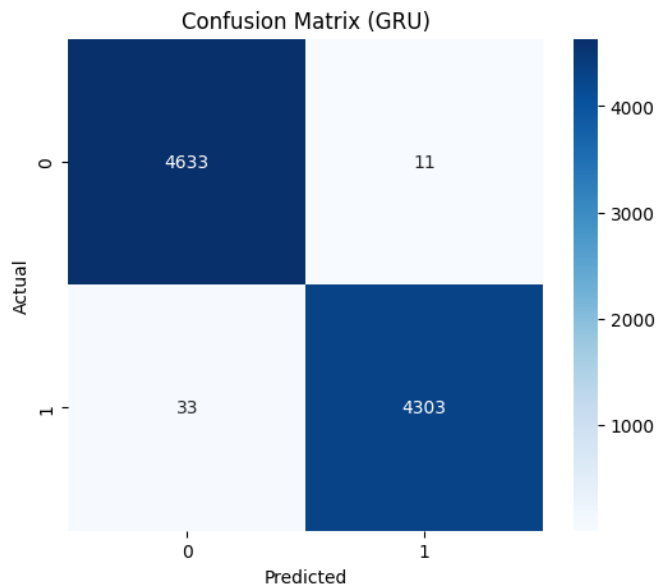
To capture sequential patterns and long-term dependencies in text, **Recurrent Neural Networks (RNNs)** were implemented:

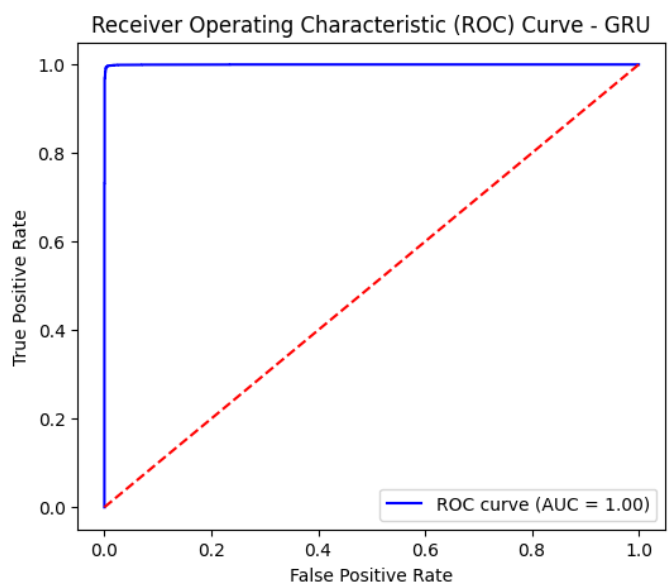
- **RNN:** Basic recurrent architecture for sequential data.



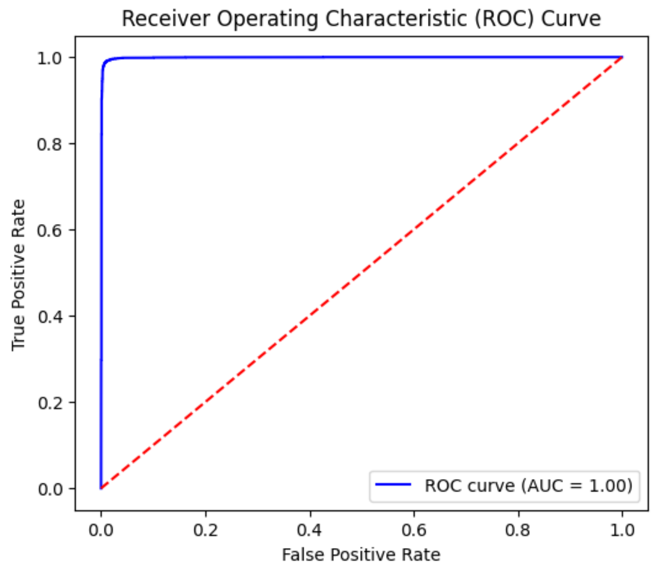


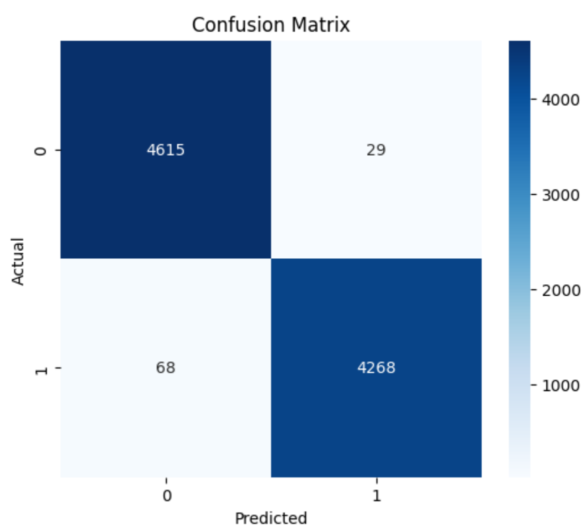
- **GRU (Gated Recurrent Unit):** More efficient variant of RNN, better at handling vanishing gradients.





- **LSTM (Long Short-Term Memory):** Captures long-term dependencies effectively, improving performance on longer articles.





	precision	recall	f1-score	support
0	0.99	0.99	0.99	4644
1	0.99	0.98	0.99	4336
accuracy			0.99	8980
macro avg	0.99	0.99	0.99	8980
weighted avg	0.99	0.99	0.99	8980

Sequential models were trained on tokenized and padded sequences, with pretrained embeddings (Word2Vec, GloVe, FastText) to improve semantic understanding.

---

5. Transformer-Based Models

State-of-the-art **Transformer models** were applied to maximize classification accuracy:

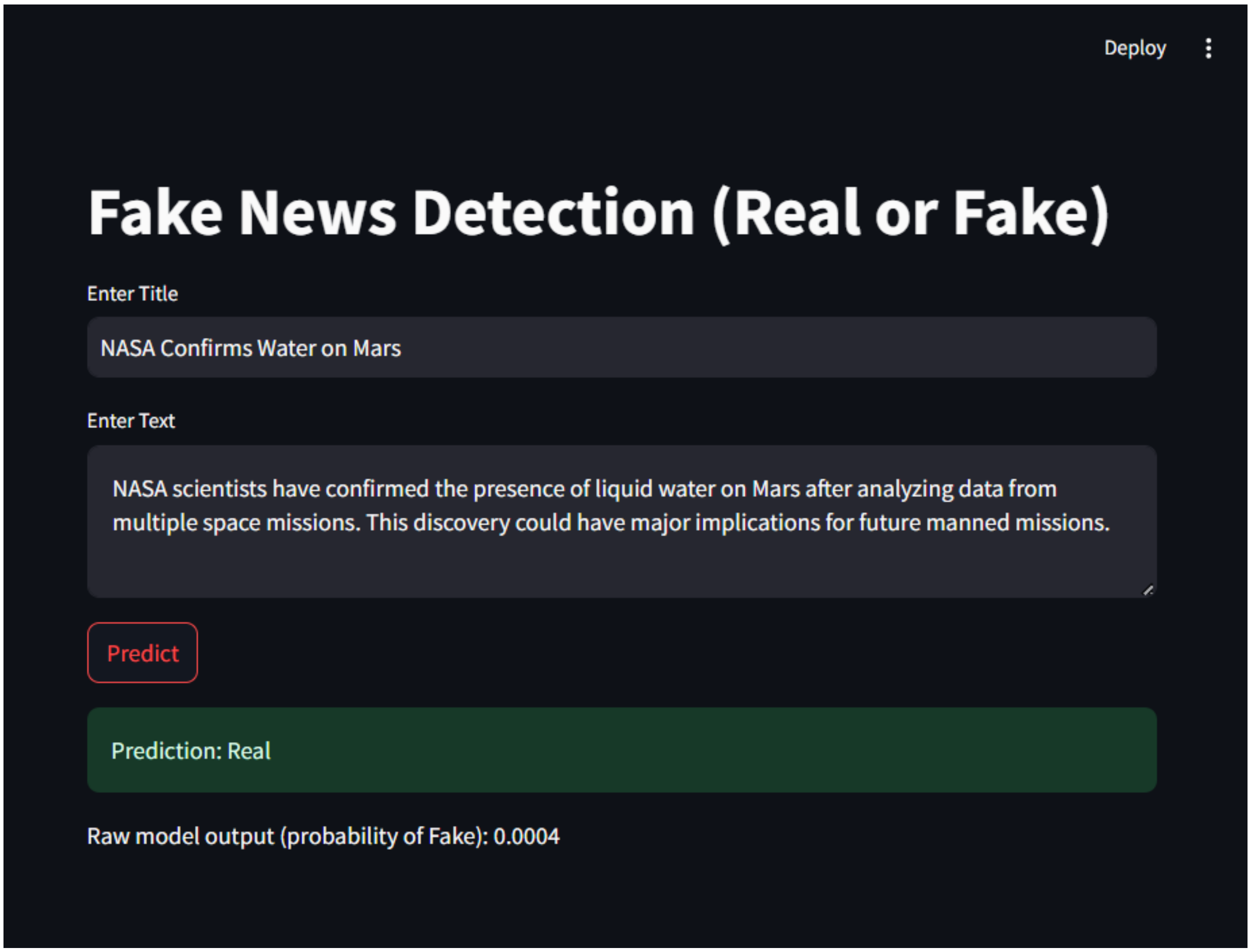
- **BERT (Bidirectional Encoder Representations from Transformers):** Pretrained on a large corpus and fine-tuned on the dataset.

	precision	recall	f1-score	support
0	0.84	0.92	0.88	3213
1	0.92	0.84	0.88	3522
accuracy			0.88	6735
macro avg	0.88	0.88	0.88	6735
weighted avg	0.88	0.88	0.88	6735

---

7. Deployment

The final pipeline is deployed using **Flask**, providing a user-friendly interface where users can input a news headline or article and receive real-time classification as **fake or real**. This demonstrates practical applicability and supports business decision-making by reducing manual verification costs.



Ethical & Societal Considerations

While this project focuses on building a robust fake news detection pipeline, careful attention was given to the **ethical and societal implications** of deploying such systems.

1. Dataset Biases

- The dataset contains news from different sources, and biases may exist in terms of **political leaning, subject focus, or language style**. For example, certain political topics may appear more frequently in fake news, which could influence model predictions.
- It is crucial to acknowledge that the model may inadvertently **reflect these biases**, potentially affecting the fairness and neutrality of predictions.

2. Importance of Transparency

- Understanding why the model classifies an article as fake or real is essential for trust and accountability.
- Transformer models, such as BERT, allow some degree of **interpretability** through attention mechanisms, helping explain which words or phrases influenced the prediction.
- Transparency is particularly important when these tools are used in **news verification or content moderation**, as users and stakeholders need to understand the reasoning behind automated decisions.

3. Responsible Use of AI

- The system should be used to **assist human reviewers** rather than fully replace them, ensuring that final decisions consider context and nuance.
- Automated fake news detection can reduce **resource consumption and operational costs** by flagging potentially misleading content early, but misuse could lead to **censorship or unjustified suppression** of legitimate news.
- Guidelines and policies should accompany deployment to ensure AI tools support **ethical content moderation**, public awareness, and societal benefit.

Conclusion:
Ethical considerations are integral to deploying fake news detection systems. Addressing dataset biases, maintaining transparency in predictions, and promoting responsible AI use are critical to ensuring that the technology **enhances trust, reduces misinformation, and operates fairly** in real-world applications.