Machine Learning LAB

PROJECT REPORT

on

Fake News Detection using NLP and BERT

Submitted By:

RAHUL KINDO (121CS0178)

NEELAMSETTI HARIKRISHNA (121CS0179)

MD NAFIS AL SAFAYET (121CS0217)

Submitted To:

Prof. Ratnakar Dash

Department of Computer Science and Engineering

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA

14 November 2024

INDEX

ABSTRACT:

The project appears to cover the analysis of a dataset containing both "fake" and "true" news articles. It begins with loading necessary libraries (NumPy, pandas, and matplotlib) followed by loading CSV files containing fake and true news data into pandas Data Frames. The analysis includes data exploration, such as examining data shapes and columns, balancing the dataset, and visualizing word frequencies through word clouds for both fake and true news texts. This preliminary analysis aims to highlight distinguishing textual patterns between fake and true news content, possibly as part of a broader effort to build a classification model or conduct a more detailed text analysis. A custom BERT layer is created using TensorFlow's Keras framework, which involves defining a Bert Layer class that integrates BERT embeddings within a larger neural network. This BERT-based architecture is likely used to generate text embeddings that feed into a classifier to predict whether the news is fake or true. Using BERT in this way leverages its pre-trained language understanding, which can improve the accuracy and depth of text classification models for nuanced tasks such as fake news detection.

ABBREVIATIONS:

- **BERT –** Bidirectional Encoding Representations from Transformers
- **NLP –**Natural Language Processing
- **ROC –** Receiver Operating Characteristic
- **AUC–** Area Under Curve
- **CSV–**Comma-Separated Values

List of Figures:

1. Introduction

In today's digital landscape, the rapid spread of information has paradoxically facilitated the rise of fake news—misleading information presented as credible news—that can distort public perception and undermine trust in legitimate journalism. As misinformation increasingly becomes a tool for manipulation, the need for effective fake news detection methods has grown paramount. This report explores various strategies and technological advancements deployed to combat fake news, ranging from machine learning algorithms to collaborative fact-checking initiatives. By examining the current state of fake news detection, this study aims to shed light on the challenges faced in identifying and mitigating misinformation and the crucial role technology plays in fostering a more informed society.

2. Literature Review

This notebook project is focused on distinguishing between fake and true news articles, a topic of increasing importance in the digital age. It begins with data loading, exploratory data analysis, and preprocessing, including dataset balancing and word cloud visualizations. These initial steps offer insights into the dataset's structure and highlight key vocabulary differences, aiding in the downstream classification tasks.

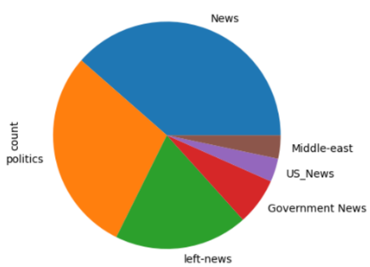
This notebook utilizes BERT (Bidirectional Encoder Representations from Transformers), a pre-trained language model known for its ability to capture contextual relationships between words in a text. By employing BERT, the notebook capitalizes on its transformer-based architecture, which considers both left and right word contexts in every sentence. The BERT tokenizer and model, loaded through the Hugging Face library, enable efficient tokenization and transformation of raw text data into embeddings that preserve semantic meaning.

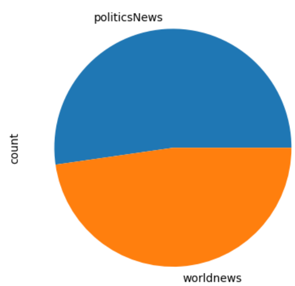
This layer setup provides the model with rich representations of each article, enhancing the classification's ability to discern the subtle language cues that may indicate fake versus true news. Additionally, the custom BERT layer constructed in TensorFlow Keras integrates these embeddings into a broader neural network architecture, a technique widely noted in recent literature for improving classification accuracy on text-based tasks. .

3. PROPOSED METHODOLOGY

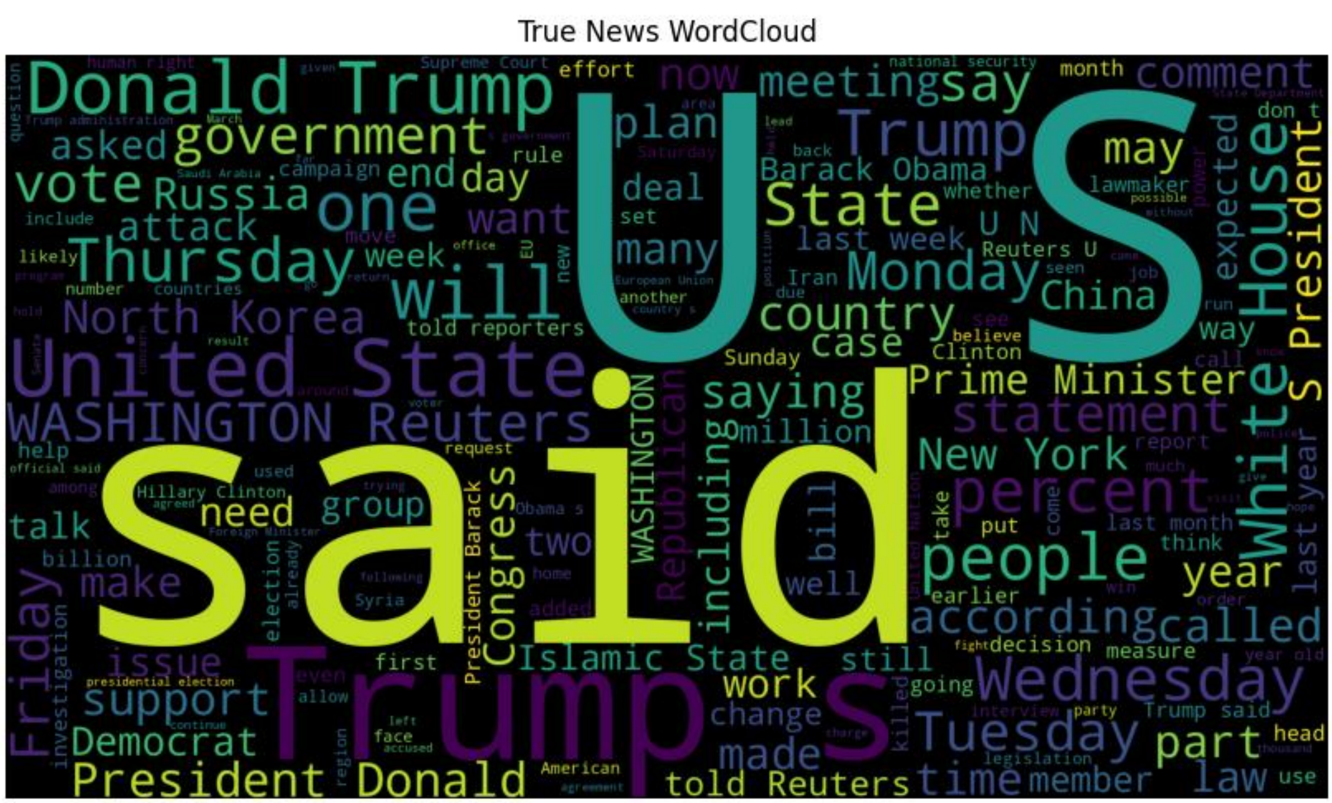
3.1 Dataset Overview

The dataset used in this project is the Fake News detection dataset on Kaggle website. Kaggle's fake news detection dataset typically includes two main labeled subsets: "True" and "Fake" news articles, each with fields like title, text, subject, and date. The **title** and **text** provide key language features, while **subject** (e.g., "politics") and **date** add contextual depth when available. These datasets are often balanced between true and fake articles, enabling effective supervised learning and reducing bias during model training. Widely used in NLP and machine learning projects, the dataset supports the development of classification models ranging from traditional algorithms to advanced techniques like BERT.





Word Cloud for True News



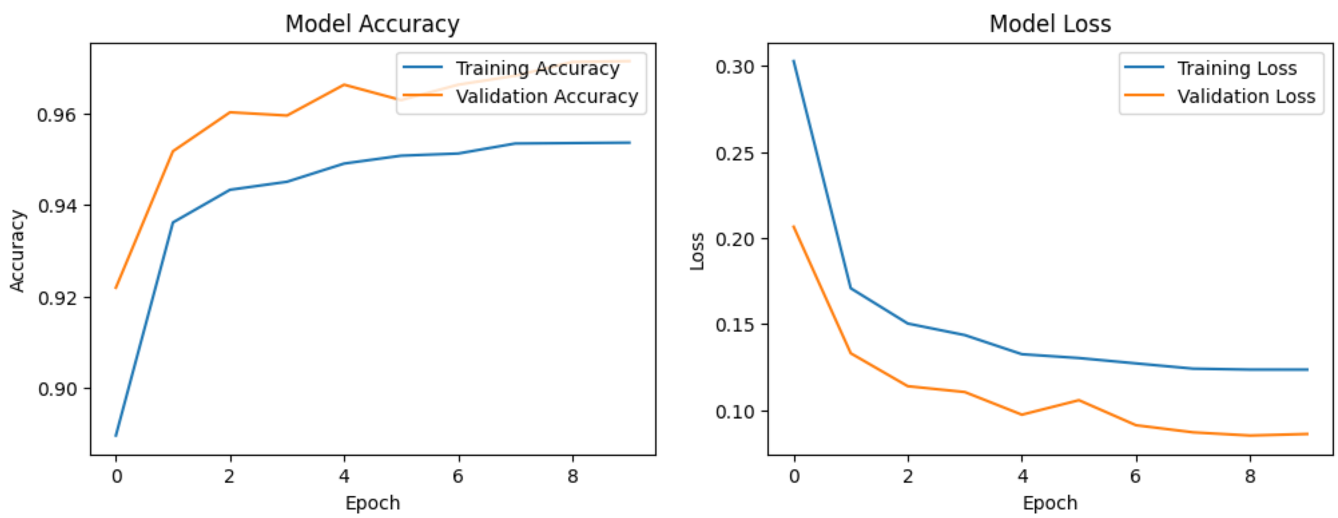
Word Cloud for Fake News

3.2 Data Preprocessing

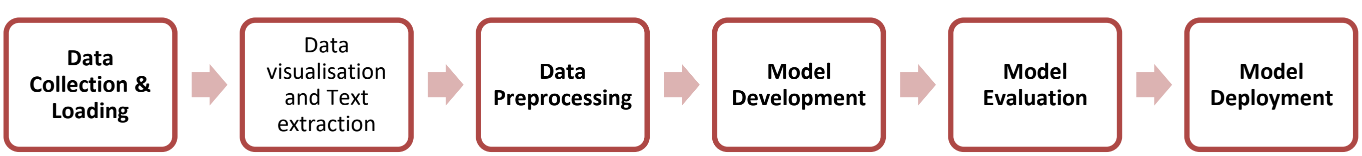
Data preprocessing function transforms a sentence for text analysis by applying several steps. First, it converts the text to lowercase for uniformity, then tokenizes it into individual words. It removes common stopwords (like "the" or "is") and applies stemming, reducing words to their root forms (e.g., "running" to "run"). Afterward, the words are joined back into a sentence, and an NLP model is applied for lemmatization, which converts words to their dictionary forms (e.g., "better" to "good"). Finally, the lemmatized words are rejoined into a single processed sentence, ready for analysis.

3.3 Model Development and Training

Model trains a neural network model for classifying text, potentially to detect fake news. First, it uses BERT for preprocessing and encoding, where Bert preprocess tokenizes and prepares text input, and Bert encoder converts this processed text into embeddings. The model adds dropout layers to prevent overfitting and a dense layer with 32 units and a sigmoid activation for feature extraction, followed by another dropout layer and a final dense layer with a sigmoid activation to output a binary classification (likely real or fake news). The model is compiled with the Adam optimizer, binary cross-entropy loss (suitable for binary classification), and accuracy as a metric. It's trained on the fake news dataset with a batch size of 64 for 10 epochs, using separate training and validation sets to monitor performance during training.



3.4 Block Diagram of Proposed Method:



4. RESULTS

4.1 Performance Metrics

Accuracy: Calculated as (TP +TN) / (TP + TN +FN + FP), where TP is true positives and FN is false negatives, TN is true negatives and FP is false positives

Sensitivity/Recall: Calculated as TP / (TP + FN), where TP is true positives and FN is false negatives.

Specificity: Calculated as TN / (TN + FP), where TN is true negatives and FP is false positives.

Precision: Calculated as TP / (TP + FP), where TP is true positives and FP is false positives.

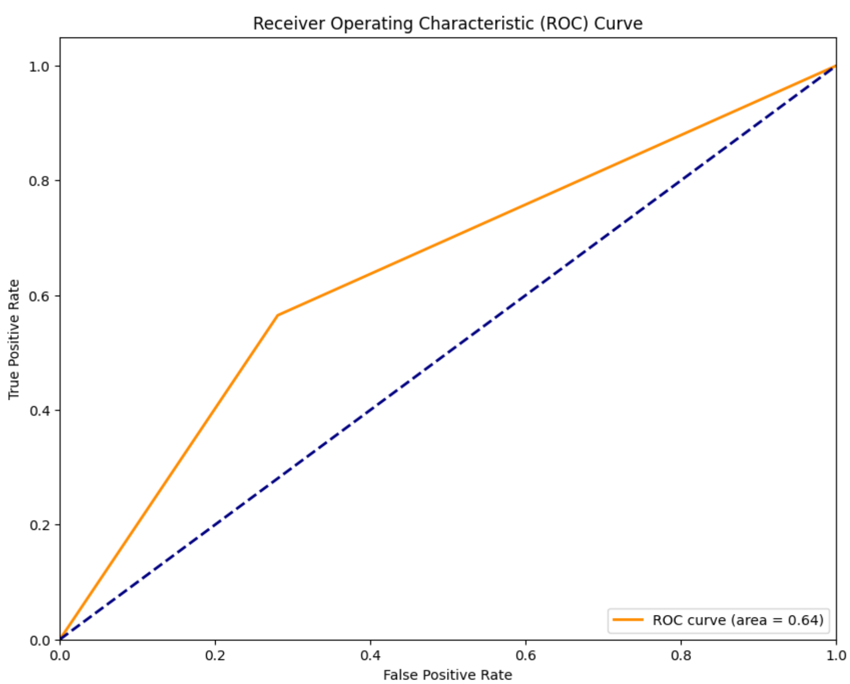
F1 Score: Calculated as hormonic mean of precision and recall

Confusion Matrix: Shows the distribution of true positive, true negative, false positive, and false negative predictions.

4.2 Results

Using Pre-trained BERT model

ROC-AUC curve:



Using custom BERT model

Test Accuracy: The model achieved an accuracy of 0.97.

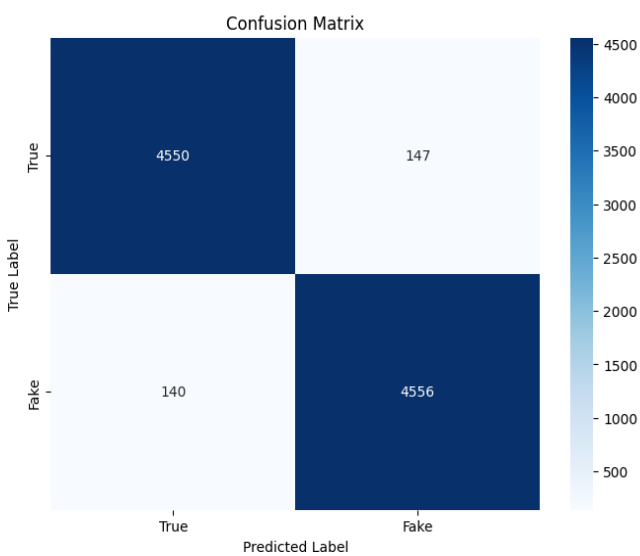
Sensitivity: The model achieved a sensitivity of 0.97.

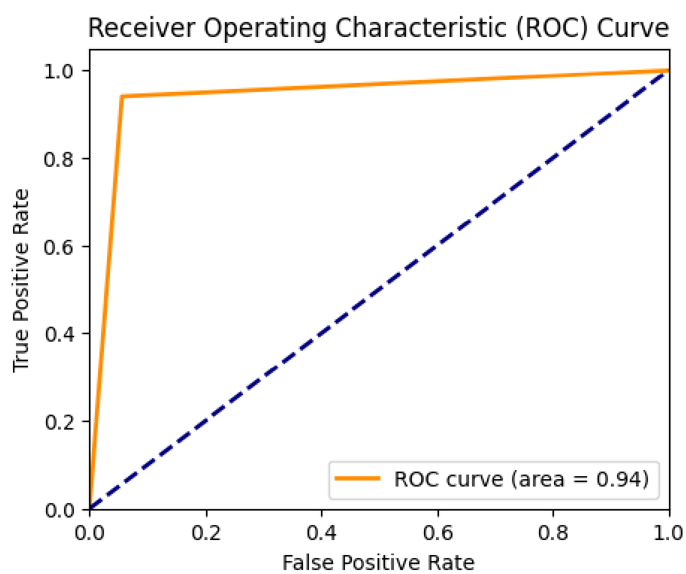
Specificity: The model achieved a specificity of 0.96.

Precision: The model achieved a specificity of 0.97.

F1 Score: The model achieved a f1 score of 0.97.

Confusion Matrix and ROC-AUC curve:





5.CONCLUSION

In this project we successfully developed a sentiment analysis model on Amazon Alexa reviews to classify reviews as positive or negative. After thorough data preprocessing, including text cleaning, stopword removal, and feature extraction using CountVectorizer, various machine learning models were trained. Techniques like SMOTE were applied to address class imbalance. Among the models tested, Random Forest performed best, achieving high accuracy and generalization on unseen data. The project demonstrated effective sentiment prediction, providing valuable insights into customer feedback. Future improvements could include experimenting with advanced deep learning models like LSTM for even better performance on larger dataset.

6.GUI development:

I have developed a Flask web application that allows users to enter the title and text of the news, which is then processed by a machine learning model to predict whether the given news is fake or true.

Steps to Run:

1. Set up the environment:

    - Ensure Python and Flask are installed. You can install Flask via pip install flask.
    - Ensure all necessary dependencies are installed, including any deep learning libraries (like TensorFlow or Keras) used for model inference.
    - Run the Flask app:
    - Navigate to the project directory in the terminal and run the Flask application with python app.py.
    - The app should be available at http://127.0.0.1:5000/ in your browser.

6.References:

1. Hunt Allcott and Matthew Gentzkow. 2017. Social media and fake news in the 2016 election. Technical Report. National Bureau of Economic Research.
2. Michael Barthel, Amy Mitchell, and Jesse Holcomb. 2016. Many Americans believe fake news is sowing confusion. Pew Research Center Vol. 15 (2016).
3. Moraes, R., Valiati, J. F., & Gaviao Neto, W. P. (2013). Document-level sentiment classification: An empirical comparison between SVM and ANN. Expert Systems with Applications, 40(2), 621–633.
4. M. Yousefi-Azar and L Hamey, "Text summarization using unsupervised deep learning", *Expert Systems with Applications*, vol. 68, pp. 93-105, 2017.
5. Derek Miller, "Leveraging BERT for extractive text summarization on lectures", *arXiv preprint arXiv:1906.04165*, 2019.