

Fake News Detection System

Bhuwan Mishra, Gagan Kaushik, Kshitij Johri, Arun Kumar Prasad, Vijaylakshmi Sajwan

Btech-Student, CSE, Shivalik College of Engineering, Dehradun, Uttarakhand, India,
mishra.bhuwan104@gmail.com

Btech-Student, CSE, Shivalik College of Engineering, Dehradun, Uttarakhand, India,
gagan90815@gmail.com

Btech-Student, CSE, Shivalik College of Engineering, Dehradun, Uttarakhand, India,
kshitijjohri0@gmail.com

Btech-Student, CSE, Shivalik College of Engineering, Dehradun, Uttarakhand, India,
aadityaprasad37@gmail.com

Professor, B-tech, Shivalik College of Engineering, Dehradun, Uttarakhand, India,
vijaylakshmi.sajwan@sce.org.in

Abstract: With technological advancements, social media has been used as one of the tools for information propagation. Unfortunately, the increase in fake news in the society has led to various social, political, and economic challenges. This paper discusses the development of an automatic system to detect fake news based on machine learning and NLP approaches. News articles are classified into fake or real classes based on their texts.

Several NLP processes have been used in this research including text pre-processing, tokenizing, removing stop words, and extracting features by applying TF-IDF. Furthermore, BERT has been selected as the deep learning approach to represent contextualized features using the RoBERTa tokenizer. Data will be divided into training, validation, and testing datasets (70%, 15%, and 15% respectively). For model optimization, AdamW optimizer will be used with a learning rate of $2e-5$ and batch size 16 with training being done for three to five epochs.

The performance of the model is evaluated using accuracy, precision, recall, F1-Score, and a confusion matrix.

Keywords: *Fake News Detection, BERT, Natural Language Processing, TF-IDF, Machine Learning*

I. INTRODUCTION

1.1 Project Overview

In recent years, the internet and social media have rapidly grown, leading to a significant increase in how information spreads. While this has improved communication and access to news, it has also resulted in the widespread sharing of fake news. Fake news is false or misleading information presented as legitimate news. It can influence public opinion, create confusion, and even lead to social and political instability.

The Fake News Detection project is a machine learning application that classifies news articles into two categories: real and fake. The system uses a labeled dataset of news articles to train and test the model. The workflow starts with data preprocessing, where the text data is cleaned and prepared for analysis. This is followed by feature extraction using TF-IDF and contextual representation with BERT embeddings.

1.2 Problem Statement

In today's digital age, the rapid growth of online platforms and social media has made it very easy to share information instantly. This convenience has led to the widespread spread of fake news, which is often misleading, inaccurate, or completely false. Such misinformation can influence public opinion, create panic, harm reputations, and affect political and social stability.

Detecting fake news manually is hard because of the huge amount of content generated every day. Traditional methods

are not effective enough to verify the authenticity of news in real-time. This creates a strong need for an automated system that can accurately identify and classify news articles as real or fake using machine learning and NLP techniques.

1.3 Motivation and Justification

This project aims to tackle the negative effects of fake news on society. By using a machine learning approach, it offers a scalable and efficient way to identify fake news. The system employs NLP techniques to analyze text patterns and pull out important features from news content, which improves its classification accuracy. It can be adapted for various applications, including browser extensions, mobile apps, or news verification tools.

1.4 Project Objectives

The main objectives for this project include:

- Design a highly accurate fake news classifier utilizing machine learning and NLP techniques.
- Gather and consolidate datasets containing both genuine and fake news stories.
- Carry out exploratory data analysis to learn about the nature of data.
- Utilize techniques such as TF-IDF vectorization and BERT embeddings to represent features.
- Train and assess different machine learning classifiers such as Logistic Regression, Naive Bayes, SVM, and BERT.
- Store the model for predictions without needing re-training.

II. LITERATURE REVIEW AND RELATED WORK

2.1 Background Theory

Fake News Detection refers to the identification of false or manipulated news articles through computational methods. It is underpinned by machine learning (ML) and natural language processing (NLP) techniques. ML techniques allow the system to learn features from labeled data and make predictions, with supervised ML being highly effective in classifying fake news articles through the use of labeled datasets comprising actual and fake articles.

NLP techniques are key in dealing with the initial data since data in the form of unprocessed text cannot be input into an ML model. Tokenization, stop-word elimination, normalization, and other preprocessing techniques help clean up the dataset and enhance its quality. Feature extraction techniques such as TF-IDF allow for converting the text into feature vectors according to the importance of words in relation to the whole dataset.

2.2 Review of Related Work

Detection of fake news has been a topic of research interest using a variety of ML and deep learning techniques:

- Castillo et al. (2015): Suggested one of the early techniques to detect credibility of social media information through supervised ML algorithms and feature engineering.
- Shu et al. (2016): Conducted analysis of fake news with respect to social context like user behaviors and fake news propagation on social media.
- Wang (2017 – LIAR Dataset): Created a fake news detection benchmark dataset and utilized Logistic Regression and SVM classifiers.
- Rashkin et al. (2018): Investigated language characteristics in texts using NLP and ML models to classify texts.
- Zhou et al. (2019): Applied deep learning techniques including RNN and LSTM to capture contextual relationships from textual information.
- Kaliyar et al. (2020): Used combined approach of ML and deep learning methods to improve classification of large datasets.
- Devlin et al. (2021 – BERT): Developed transformer architectures to capture contextual meanings and enhance fake news detection.

- 2022 – Present: Researchers concentrate on real-time detection systems, efficient models, and applications such as browser extensions and automatic verification.

2.3 Project Contribution

The major contribution to fake news detection made by this project lies in designing a comprehensive ML pipeline covering data unification, EDA, feature extraction, model building, and evaluation. Significant contributions include: (1) employing both traditional ML techniques and BERT-based deep learning for performance comparison; (2) utilizing TF-IDF vectors and BERT embeddings for feature generation; (3) saving trained models with vectorizer for instant predictions; and (4) evaluating model performance based on accuracy, precision, recall, F1-score, and confusion matrix.

III. METHODOLOGY AND PROPOSED SOLUTION

3.1 Environment Setup

Development environment setup includes Python 3.X with virtual environment (venv or anaconda), installation of necessary packages (numpy, pandas, sklearn, matplotlib, seaborn, pytorch, transformers), project structure creation in working directory, and processing data through `data_unification.py` script with exploratory analysis using `eda.py`. Trained model and vectorizer are saved as `fake_news_model.pkl` and `vectorizer.pkl` respectively.

3.2 Dataset Reading and Unification

The real and fake news datasets are imported using the pandas library from their respective CSV files. Both datasets are combined into a single unified dataset containing a label field where '1' means real news and '0' implies fake news. All irrelevant columns are dropped, only text fields are kept. Duplicates are deleted, missing values are treated, and text formatting is done to prepare the data.

3.3 Dataset Overview and Analysis

The EDA (Exploratory Data Analysis) is used to understand the structure of the data, type of features, and distribution of classes. The main textual feature is the news headline, while the target feature is authenticity. Text length analysis is performed by determining word/character count for each news article and analyzing it through histograms. Differences in text length of real vs. fake news are determined.

3.4 Data Preprocessing

The process of preprocessing textual data involves the following steps:

- Conversion of all texts into lower case.
- Removal of punctuation symbols.
- Exclusion of commonly occurring stop words which do not help in predicting.
- Tokenization, i.e., separation of texts into tokens.
- Stemming/Lemmatization of texts.
- Division of dataset into 70%, 15% and 15% for training, validation and testing, respectively.

3.5 Feature Representation

Two types of feature extraction are used for this project. First is TF-IDF (Term Frequency-Inverse Document Frequency), where numeric vectors are created by giving weights to terms according to how often they appear in relation to the whole document set; applied using classical ML models.

The second type is BERT (Bidirectional Encoder Representations from Transformers), which extracts features from a deep contextual perspective. Using the RoBERTa tokenizer, each news headline is tokenized into subwords, token IDs are extracted and attention masks are generated to differentiate between valid and padded tokens, with 15 tokens chosen as the maximum sequence length based on EDA findings.

3.6 Model Building Strategy

The pre-trained BERT layers are frozen to cut down computation expenses and avoid overfitting. A classification head is included at the end of the BERT layers by adding a dense layer. The [CLS] token output serves as input to the classifier. BCE Loss is utilized for binary classification between real and fake news. AdamW is used as optimizer with a learning rate of $2e-5$.

III. IMPLEMENTATION DETAILS

4.1 Tools and Technologies

- Python: Data preprocessing, model building, training, and evaluation.
- PyTorch: Model building, training, and evaluation using BERT-based neural networks.
- Hugging Face Transformers: Pre-trained BERT models and RoBERTa tokenizer.
- Pandas and NumPy: Dataset handling and preprocessing.
- Matplotlib and Seaborn: Visualizations for class distribution and text lengths.
- Scikit-learn: Dataset splitting, traditional ML algorithms, and evaluation metrics.
- Google Colab: Development platform with integrated GPU.
- Flask: Web app that predicts real-time news from input.
- GitHub: Project management and version control system.

4.2 Training Process

The training architecture uses Optimizer: AdamW; Learning rate: $2e-5$; Batch size: 16; Epochs: 3-5. The loss value is tracked during training to make sure the model is indeed learning. Validation metrics are computed at the end of every epoch to watch out for overfitting. Baseline comparisons are done with Logistic Regression, Naive Bayes, and Support Vector Machines.

4.3 BERT Fine-Tuning

The pre-trained model of BERT is loaded through Hugging Face Transformers. Input data in tokenized form is fed into the model. BCE loss function is computed, and then backpropagation is used to update the weights of the classifier head without updating the weights of the BERT layers.

4.4 Advanced Enhancements

Further improvements include:

- (1) Metadata Features – author's name, publishing date, and source are included for additional context;
- (2) Social Features – shares, likes, comments, and interactions for assessing news credibility;
- (3) Attention-based Fusion – text features combined with metadata and social features via attention-based approach;
- (4) Explainability – SHAP and LIME approaches applied for explaining predictions by identifying key words contributing to labeling news as real or fake.

IV. TESTING

Testing is done to find out if the fake news detection system works properly and effectively. The types of tests done include:

- Unit Test: Each module of the system—preprocessing, feature extraction, and prediction—are individually tested.
- Integration Test: The entire system is tested for seamless functioning and effective communication between all modules.
- System Test: Testing of the entire system.
- Accuracy Test: The machine learning model is tested using the test set.
- User Interface Test: Testing of the Flask user interface.

TABLE I

Test Case	Input	Expected Result	Status
Valid Real News	Real news article	Predicts: Real	Pass
Valid Fake News	Fake news article	Predicts: Fake	Pass
Empty Input	Blank text	Shows warning/error	Pass
Large Content	Long article text	Processes successfully	Pass

TEST CASES

V. RESULTS AND EVALUATION

6.1 Performance Evaluation

Several metrics are used to evaluate the BERT fake news detection system. Accuracy is the main metric used to measure how well the model can detect fake news by calculating the proportion of articles correctly identified. Other performance measures include precision, recall, and the F1 score.

A confusion matrix provides an assessment of true positive, true negative, false positive, and false negative parameters, giving insights about the model's ability in classifying real news vs. fake news. Training loss is computed at each epoch, and the model learns when there is always a decrease in loss.

TABLE II

MODEL PERFORMANCE COMPARISON

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Naive Bayes	82.4	80.6	83.1	81.8
Logistic Regression	87.3	86.1	88.0	87.0
SVM	89.5	88.7	90.2	89.4
BERT-based (Proposed)	95.8	95.1	96.3	95.7

6.2 Discussion of Results

The obtained result highlights the high competence of the BERT-based model when performing classification of news articles. The model exhibits accuracy of 95.8% and F1-score of 95.7%, which is noticeably higher than Naive Bayes (82.4%), Logistic Regression (87.3%), and SVM (89.5%). A steady drop in training loss values throughout all training stages proves an efficient learning process. Additionally, stability of validation metrics indicates the absence of overfitting.

The confusion matrix analysis shows that the model successfully recognizes the vast majority of real and fake news articles. High performance of BERT compared to bag-of-words methods is caused by its ability to provide contextual and semantic understanding of texts. Further improvements are possible through utilization of metadata and social network information.

VI. CONCLUSION

This paper efficiently implements a BERT-based model for identifying fake news that classifies news articles into real and fake categories using sophisticated methods of natural language processing and machine learning. The model has gone through all stages of development including data collection and integration, preprocessing, EDA, transformation with TF-IDF and BERT, model training, and evaluation.

The experiment shows that the implemented BERT-based classifier performs significantly better than baseline ML models, achieving 95.8% accuracy and 95.7% F1 score. The context understanding capability of BERT was a decisive factor behind such an impressive result. Future improvements could involve metadata and social features, advanced transformers (RoBERTa, XLNet, GPT), larger training datasets, and deployment as a web or mobile application.

VIII. FUTURE SCOPE

- Real-time news validation from news sources and social media platforms.
- Multi-language capability to enable detection of fake news in regional languages.
- Development of a mobile app.
- Implementation of fake image and video detection with computer vision algorithms.
- Use of bigger and ever-growing datasets to ensure better generalization.
- Integration of sentiment and behavior analysis for better detection.
- Deployment in cloud environment (AWS, GCP, Azure) for wide deployment.
- Development of a browser extension for automatic news validation.
- Application of more sophisticated architectures such as RoBERTa, XLNet, and GPT.
- Use of explainable AI methods like SHAP and LIME.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to Dr. Vijaylakshmi Sajwan for her valuable suggestions and constant encouragement. We also wish to thank all other faculty members and staff of the Department of Computer Science and Engineering for their help in completing this project successfully.

REFERENCES

- [1] Allcott, H., & Gentzkow, M. (2017). Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives*, 31(2), 211-236. <https://doi.org/10.1257/jep.31.2.211>
- [2] Ruchansky, N., Seo, S., & Liu, Y. (2017). CSI: A Hybrid Deep Model for Fake News Detection. In *Proceedings of the ACM Conference on Information and Knowledge Management*. <https://doi.org/10.1145/3132847.3132877>
- [3] Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2018). Fake News Detection on Social Media: A Data Mining Perspective. *ACM SIGKDD Explorations Newsletter*. <https://doi.org/10.1145/3219819.3219900>
- [4] Perez-Rosas, V., Kleinberg, B., Lefevre, A., & Mihalcea, R. (2018). Automatic Detection of Fake News. In *Proceedings of COLING*. <https://doi.org/10.18653/v1/C18-1287>
- [5] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL*. <https://doi.org/10.18653/v1/N19-1423>
- [6] Kaliyar, R. K., Goswami, A., & Narang, P. (2021). FakeBERT: Fake News Detection in Social Media with a BERT-based Deep Learning Approach. *Multimedia Tools and Applications*. <https://doi.org/10.1109/ICMLA51294.2020.00145>
- [7] Hugging Face. (2024). Transformers Library Documentation. <https://huggingface.co/docs>
- [8] Scikit-learn Developers. (2024). Model Evaluation Metrics. <https://scikit-learn.org/stable/>

GitHub Repository: https://github.com/Gagan2004/Fake_News_Detector