BERT-Based Fake News Detection System with Comparative Analysis of Classical Machine Learning Models

Uday Singhal
Department of Computer Science and Engineering
Delhi Technological University
Rohini, Delhi
udayalwayshere@gmail.com

Tanishq Bhardwaj
Department of Computer Science and Engineering
Delhi Technological University
Rohini, Delhi
tanishqbhardwaj457@gmail.com

***Abstract*— The proliferation of misinformation in digital media has led to a pressing need for effective fake news detection systems. This study presents a comprehensive approach to detecting fake news using both classical machine learning models and a transformer-based BERT model. The models were trained and tested on a labeled dataset of news articles with attributes such as title, description, and label (real or fake). The BERT model achieved an accuracy of 97.47%, outperforming all classical models. Additionally, this research emphasizes model interpretability, scalability, and real-world deployment aspects**

Keywords: Fake News, BERT, Natural Language Processing, Machine Learning, Text Classification, Transformer Models

I. INTRODUCTION

Fake news poses a significant threat to societal trust and democratic discourse. With the rise of social media and online news platforms, misinformation can spread rapidly and reach large audiences. Manual verification methods are not scalable, highlighting the need for automated detection systems. This study explores automated fake news detection using Natural Language Processing (NLP) techniques. By employing both classical machine learning and modern transformer-based models like BERT, we aim to identify the best-performing technique in terms of accuracy and scalability. Real-time deployment and the ability to adapt dynamically to new forms of misinformation are also key aspects addressed in this research. Furthermore, the use of contextual embeddings from BERT enables more nuanced understanding of language than traditional vectorization techniques. Our research contributes not only an implementation perspective but also addresses practical concerns like interpretability and real-world robustness.

II. RELATED WORK

Numerous studies have explored fake news detection through machine learning. Traditional models like Logistic Regression, Naive Bayes, and Support Vector Machines (SVM) have been applied to datasets such as LIAR, ISOT, and FakeNewsNet. These approaches utilize handcrafted features and bag-of-words representations. Deep learning models like LSTM and CNN have also demonstrated promise by capturing sequential patterns in text. Recently, transformer-based architectures like BERT have emerged as powerful tools for contextual text representation, outperforming older models across various NLP tasks. Transfer learning using BERT has shown significant improvements in various domains of text classification. Our work builds on this foundation by implementing and comparing these techniques on a labeled dataset, and further evaluating them in terms of their deployment feasibility, interpretability, and real-time application accuracy. Unlike some previous works which focused solely on accuracy, we also assess model scalability, visualization of predictions, and implementation ease for practical use.

III. METHODOLOGY

3.1 Dataset

The dataset used includes 6,000 labeled news entries with four columns: ID, title, description, and label (real or fake). Data preprocessing involved removing null values, duplicates, special characters, and performing lowercasing and stopword removal. Exploratory data analysis was also conducted to understand class distribution and frequent n-grams.

3.2 Classical ML Models

We used TF-IDF vectorization to convert text data into numerical format. Models like Logistic Regression, Naive Bayes, SVM, and Passive Aggressive were implemented. Example snippet:

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import PassiveAggressiveClassifier
vectorizer = TfidfVectorizer(stop_words='english', max_df=0.7)
X_train = vectorizer.fit_transform(train_data)
X_test = vectorizer.transform(test_data)
model = PassiveAggressiveClassifier()
model.fit(X_train, y_train)
predictions = model.predict(X_test)
```

3.3 BERT Model

A pre-trained BERT model (bert-base-uncased) was fine-tuned for binary classification. The model was trained for 2

epochs on GPU (T4 x2), using HuggingFace Transformers and PyTorch:

```
from transformers import BertTokenizer, BertForSequenceClassification, Trainer, TrainingArguments
from datasets import load_dataset
dataset = load_dataset('csv', data_files='news.csv')
tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
def tokenize(batch):
    return tokenizer(batch['text'], padding=True, truncation=True)
tokenized = dataset.map(tokenize, batched=True)
model = BertForSequenceClassification.from_pretrained("bert-base-uncased")
training_args = TrainingArguments(
    output_dir="./results",
    evaluation_strategy="epoch",
    num_train_epochs=3,
    per_device_train_batch_size=8,
    per_device_eval_batch_size=8
)
trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=tokenized['train'],
    eval_dataset=tokenized['test']
)
trainer.train()
```

3.4 Evaluation Metrics

Models were evaluated based on accuracy, precision, recall, F1-score, and confusion matrix. Visualization tools like 'ConfusionMatrixDisplay' and 'classification_report' from sklearn were used.

3.5 Explainability Methods

To address model interpretability, we explored SHAP (SHapley Additive exPlanations) and Integrated Gradients. These techniques help in understanding the influence of individual words or tokens on the model's prediction. SHAP assigns each word a contribution value, enabling visualizations of important features. Integrated Gradients, on the other hand, compute gradients along a path from a baseline to the input, attributing the model's prediction to specific input features.

Despite their utility, applying these methods to large transformer models like BERT posed computational and compatibility challenges. SHAP's KernelExplainer, while effective for simpler models, was prohibitively slow on high-dimensional embeddings. Similarly, Integrated Gradients required significant adaptation for token-based inputs and yielded less intuitive outputs without additional processing. Consequently, interpretability remains an ongoing challenge for complex NLP architectures, and future work will focus on refining these tools for real-time applications.

IV. Experimental Result

The following accuracies were observed:

- Logistic Regression: 91.63%
- Naïve Bayes: 89.27%
- SVM: 92.82%
- Random Forest: 91.55%
- Passive Aggressive: 92.27%
- **BERT 97.47%**

4.1 BERT Classification Report:

- Precision (Fake): 0.98
- Recall (Fake): 0.97
- Precision (Real): 0.97
- Recall (Real): 0.98

4.2 Confusion Matrix

	Predicted Fake	**Predicted Real**
Fake	611	17
Real	19	620

V. Discussion

5.1 Comparative Insights

BERT significantly outperformed all classical models in both accuracy and robustness. The transformer architecture enables better context understanding, which is crucial for detecting nuanced fake news.

5.2 Deployment Feasibility

The deployment of the model using TorchServe and integration with real-time interfaces demonstrates its practical viability. Lightweight models like Passive Aggressive may still be preferable for constrained environments.

5.3 Challenges

While BERT was able to generalize well, a few misclassifications of real-time news articles indicate the necessity of domain-specific fine-tuning and dynamic dataset updates. Interpretability methods like SHAP were explored but faced compatibility issues with large transformer models. More explainability can be achieved by integrating LIME or Captum in future iterations.

VI. Conclusion

This research demonstrates the superiority of BERT for fake news detection, offering enhanced accuracy and contextual understanding over traditional models. The model's successful deployment further signifies its potential for real-world application. Future work includes enhancing explainability using SHAP/Integrated Gradients, adapting the system for multilingual and multi-modal fake news detection, and developing dashboards for public and media organizations to utilize this tool effectively.

Acknowledgment

The author would like to thank the faculty of Delhi Technological University for guidance and support during this project.

References

[1] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." NAACL. A foundational work introducing the BERT model, which has significantly impacted NLP.

[2] Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). "Fake News Detection on Social Media: A Data Mining Perspective." ACM SIGKDD. This paper surveys machine learning techniques in detecting fake news.

[3] Vaswani, A., et al. (2017). "Attention Is All You Need." NeurIPS. Introduced the transformer architecture that BERT is based on.

[4] Pedregosa, F., et al. (2011). "Scikit-learn: Machine Learning in Python." JMLR. Details the scikit-learn library used for implementing classical ML models.

[5] Wolf, T., et al. (2020). "Transformers: State-of-the-Art Natural Language Processing." EMNLP. Describes the HuggingFace Transformers library, which was used for BERT implementation.