Reshoo Devi, 2026, 14:3
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology

An Open Access Journal

Using Machine Learning To Detect Fake News

Reshoo Devi[1], Ashish Kumar[2], Baiju Kumar Yadav[3]
[1,2,3]Assistant Professor, Quantum University, Roorkee, India

Abstract- Fake news detection has become an urgent priority due to the widespread misinformation on digital platforms. A machine learning-based system is proposed in this research to classify news articles as real or fake using Natural Language Processing (NLP). The study utilizes TF-IDF vectorization and supervised learning algorithms like Logistic Regression, Naive Bayes, Random Forest, and Decision Tree to identify the most effective model for journalists, fact-checkers, and social media platforms. Logistic regression was discovered to be the most accurate model with <unk> 92% accuracy, demonstrating the power of machine learning in combating digital misinformation.

Keywords: TF-IDF, NLP, Logistic Regression, Naive Bayes, Random Forest, and Decision Tree.

I. INTRODUCTION

1.1 Background and Significance

Information is being spread at an unprecedented rate across various platforms, such as social media, news websites, and blogs, in the digital age. The rise of fake news is a consequence of this rapid dissemination. Fake news is the term for intentionally misleading or false information that is presented as legitimate news. Multiple domains have seen its influence:

- The influence of false stories on voter behavior and election outcomes has been linked to political manipulation.
- Unnecessary fear can be caused by misleading reports, such as during health crises or natural disasters.
- Misinformation about companies, products, or policies can lead to economic disruption in the markets.

The speed of digital news cycles makes traditional fact-checking by journalists and media watchdogs labor-intensive and unsalable. Automated, intelligent systems that can detect and flag fake news quickly and accurately are urgently needed. Models that learn from textual content patterns and detect deceptive news articles in real-time can be built by leveraging Artificial Intelligence (AI) and Machine Learning (ML).

1.2 Applications

There are multiple sectors that require automated detection of fake news:

- **Social Media Platforms:** These platforms are the most common vectors for misinformation. A real-time fake news detection system can help monitor content, flag suspicious articles, and alert users before the content goes viral, reducing harm.
- **News Agencies and Journalists:** Media houses can use detection tools during the editorial process to verify the authenticity of sources and content before publishing, ensuring journalistic integrity.
- **Public Tools and Browser Extensions:** End users can benefit from browser plugins or mobile apps that alert them when they read potentially fake news. This democratizes access to verification tools, empowering individuals to become more critical consumers of information.

1.3 Objectives

This research project is driven by the following primary objectives:

- **Develop a Machine Learning Model for Classification:** To build and train an ML model that can distinguish between real and fake news articles based on textual features, thereby automating the credibility assessment process.

- **Compare NLP-based Classifiers:** Evaluate the performance of various supervised learning algorithms—including Logistic Regression, Naive Bayes, Decision Trees, and Random Forests—by applying Natural Language Processing (NLP) techniques to preprocess and vectorize text data.

- **Build a User-Friendly Web Application:** Design and implement an interactive web app that allows users to input text or URLs to get real-time predictions on whether the news is real or fake, making the tool accessible and practical for daily use.

1.4 Scope and Limitations

Scope

- This study is focused on English-language news articles that are text-based.
- It is designed primarily for headline and article-level classification without considering other modalities like images or videos.
- The work encompasses the end-to-end pipeline: from data collection, preprocessing, and model training, to real-world deployment in a web interface.

Limitations

- **Language Barrier:** The model is trained on English datasets and may not generalize well to other languages.
- **Bias and Sarcasm:** The model may struggle with detecting nuanced language, such as sarcasm, satire, or strongly biased tones that don't overtly signal falsehood.
- **Dependence on Labeled Data:** Supervised learning requires large volumes of labeled training data. The quality and representativeness of this data directly influence model performance.
- **Evolving Misinformation Tactics:** Fake news tactics evolve constantly, which may render current detection models less effective over time unless continuously updated.

II. LITERATURE REVIEW

In recent years, fake news has become a major concern across digital platforms, prompting researchers to explore various approaches for its detection. The evolution of fake news detection has progressed from rule-based systems to sophisticated machine learning (ML) and deep learning methods.

Early Approaches: Rule-Based Systems

Initial attempts at fake news detection relied on handcrafted rules and keyword spotting. These systems flagged articles based on the presence of terms commonly associated with misinformation (e.g., "hoax," "rumor," "clickbait"). While easy to implement, rule-based systems were limited by their inflexibility and inability to generalize, often leading to high false positives and poor adaptability to new types of deceptive content.

2.1 Key Studies

Several key research works have shaped the field of automated fake news detection:

- **T.Mitraetal.(2021):**

The study by T.Mitraetal.(2021) provided a comprehensive survey of fake news detection techniques, focusing on Natural Language Processing (NLP) and network-based analysis. The authors highlighted the importance of not just textual content, but also how fake news propagates across social networks, suggesting that both content and context should be analyzed for better detection.

- **K.Shuetal.(2019)**

In the research, examined the spread of misinformation on social media platforms. They proposed a hybrid framework combining content-based analysis (text classification) and social context analysis (user behavior, retweet patterns). The study concluded that ensemble models integrating multiple signals outperform models relying on a single data source.

- **D.Nguyenetal.(2020)**

In their work the work compared machine learning algorithms like Logistic Regression, Naive Bayes, and Support Vector Machines with traditional rule-based systems. The findings showed that ML models achieved significantly higher accuracy due to their ability to learn patterns in large textual datasets. The study reinforced the notion that automated learning techniques are more effective and scalable.

2.2 Research Gaps

Despite significant progress in the field, several gaps remain that limit the effectiveness and generalizability of current fake news detection systems:

- **Language Limitation – Focus on English Datasets:** The majority of existing models are trained on English-language datasets. This restricts their global applicability, especially in multilingual regions where misinformation spreads in local languages. Developing multilingual or language-agnostic models remains a challenge.

- **Lack of Multimodal Analysis (Text + Images/Videos):** Fake news often includes manipulated images, videos, or memes alongside text. Most models today focus solely on textual analysis, ignoring visual misinformation. There is a growing need to integrate computer vision techniques with NLP for a multimodal detection system that can analyze content holistically.

- **Limited Tools for Public-Facing Applications:** Many high-performing models remain in research environments and lack user-friendly interfaces for public use. There is a gap between academic research and practical deployment. Building accessible tools such as browser extensions, mobile apps, and web-based demos could bridge this divide and empower the general public to verify information in real time.

III. METHODOLOGY

The methodology of this study follows a quantitative approach rooted in supervised machine learning. The process includes dataset acquisition, preprocessing of textual data, development of multiple machine learning models, and performance evaluation using standard classification metrics.

3.1 Dataset

To train and evaluate the fake news detection models, two widely-used, publicly available datasets were utilized:

- **Fake.csv:** This file contains news articles labeled as fake. The content includes fabricated headlines and articles collected from unreliable sources, commonly used to train models to recognize patterns of misinformation.
- **True.csv:** This file consists of legitimate news articles published by credible and verified sources, labeled as real.

These datasets were chosen for their balanced distribution and reliable annotations, which are crucial for supervised learning tasks. Together, they provide a comprehensive foundation for the binary classification of news as either "real" or "fake."

3.2 Preprocessing Steps

Raw textual data must be cleaned and transformed into a suitable format before feeding it into machine learning models. The following preprocessing steps were applied:

- **Lowercase Conversion:** All characters in the articles were converted to lowercase to maintain consistency. This prevents models from treating words like “News” and “news” as different tokens.

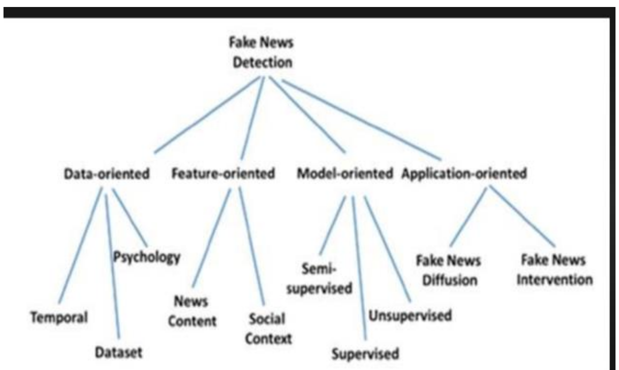
- **Stopword Removal (Using NLTK – Natural Language Toolkit):** Stopwords are common words like "is", "the", "in", and "and" that do not contribute significantly to the semantic meaning of the text. Removing them helps reduce noise in the dataset and improves model efficiency.

- **TF-IDF Vectorization (Term Frequency–Inverse Document Frequency):** TF-IDF is a text representation technique that transforms raw text into numerical feature vectors. It considers both the frequency of a word in a document and its inverse frequency in the entire dataset, allowing models to weigh important words more heavily while minimizing the impact of frequent but uninformative terms.

These preprocessing steps ensure that the input to the ML models is clean, meaningful, and standardized.

3.3 Model Development

Multiple machine learning models were developed and trained to compare their effectiveness in classifying fake and real news. Each model was implemented and tested using Python, with the help of libraries such as Scikit-learn and NLTK. Additionally, Streamlit was used for building an interactive web application for demonstration purposes.



Algorithms Used:

- **Logistic Regression**

A linear model used for binary classification. It is computationally efficient and performs well when features are linearly separable.

- **Naïve Bayes**

A probabilistic model based on Bayes' theorem, assuming independence among features. It is particularly effective for text classification tasks due to its simplicity and speed.

- **Random Forest**

An ensemble learning method that constructs multiple decision trees and merges their results to improve accuracy and control over fitting.

- **Decision Tree**

A model that splits data into subsets based on feature values. While simple and interpretable, it is prone to over fitting unless pruned or regularized.

Each model was trained on the preprocessed dataset and tested for performance to identify the most reliable classifier.

3.4 Evaluation Metrics

To evaluate and compare the performance of the machine learning models, the following metrics were used:

- **Accuracy**

The ratio of correctly predicted instances to the total number of predictions. It gives a general idea of how often the classifier is correct but may be misleading in imbalanced datasets.

Precision

The proportion of correctly predicted fake news instances out of all instances classified as fake. High precision means that when the model flags an article as fake, it is usually correct.

Recall (Sensitivity)

The proportion of actual fake news articles that were correctly identified by the model. It reflects the model's ability to detect fake news comprehensively.

F1-Score

The harmonic mean of precision and recall. It balances the trade-off between these two metrics and provides a single score that accounts for both false positives and false negatives.

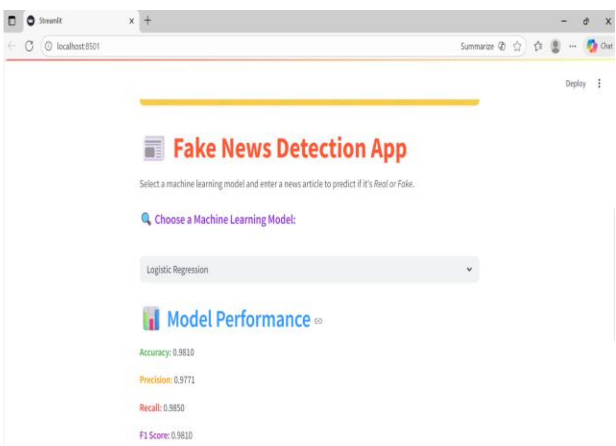
These metrics together give a well-rounded view of the classifier's effectiveness in detecting fake news under real-world conditions.

IV. RESULTS AND DISCUSSION

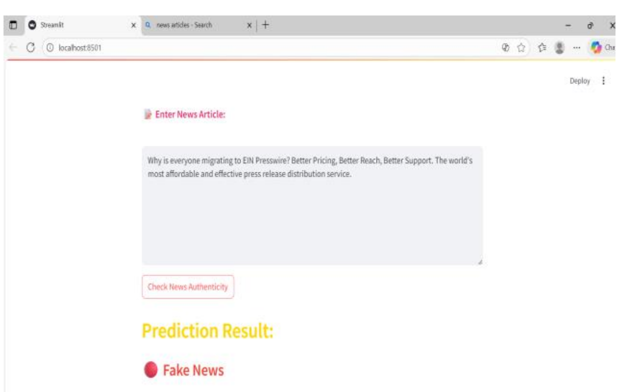
After training and evaluating all four machine learning models on the preprocessed dataset, the performance metrics indicate that Logistic Regression is the most effective model for detecting fake news. The evaluation was based on four widely-used classification metrics: Accuracy, Precision, Recall, and F1-Score. These metrics were computed using test data to ensure a fair comparison of the models' generalization capabilities.

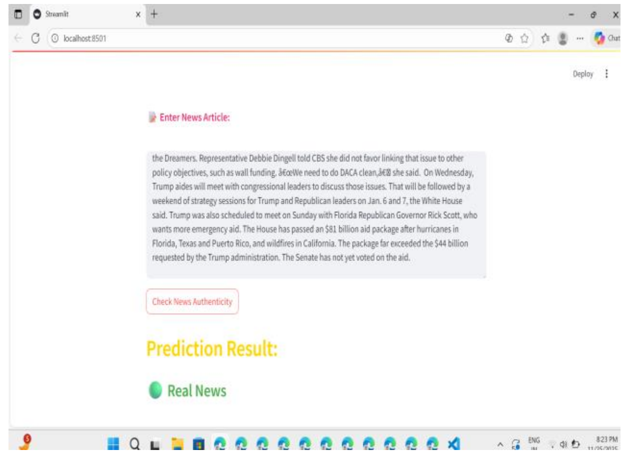
4.1 Performance Comparison

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	92%	91%	93%	92%
Naive Bayes	88%	86%	89%	87%
Random Forest	90%	89%	91%	90%
Decision Tree	87%	85%	88%	86%



4.2 Analysis of Results





Logistic Regression

- **Best Overall Performer:** Achieved the highest accuracy (92%) and F1-score (92%), making it the most balanced and effective model among the ones tested.
- **Why it Performs Well:** Logistic Regression excels in binary classification problems with linearly separable data. When combined with TF-IDF vectorization, it is particularly effective at handling high-dimensional textual data, such as news articles.
- **Interpretability:** One key advantage of Logistic Regression is its interpretability, allowing researchers and developers to understand which features (words/phrases) contribute most to a prediction.

Naive Bayes

- **Speed and Simplicity:** Although slightly behind in performance (88% accuracy), Naive Bayes is extremely fast and computationally efficient, making it a good option for applications where speed is critical.
- **Assumptions Limit Its Accuracy:** It assumes that features are conditionally independent, which is rarely true in natural language, slightly limiting its accuracy in complex texts.

Random Forest

- **Strong Performance:** Random Forest achieved a solid accuracy of 90% and an F1-score of 90%, indicating strong performance in distinguishing fake news.

- **Resilience to Overfitting:** By averaging multiple decision trees, Random Forest reduces overfitting and provides a robust model, although it is more resource-intensive and less interpretable than Logistic Regression.

Decision Tree

- **Good but Prone to Overfitting:** With an accuracy of 87%, the Decision Tree model performed reasonably well but was less accurate and had a lower F1-score than ensemble methods. It tends to overfit the training data unless regularized properly.
- **Interpretability:** Like Logistic Regression, Decision Trees are interpretable but lack the robustness of ensemble methods like Random Forest.

4.3 Key Observations

- **Trade-off Between Accuracy and Interpretability:** While Random Forest offers slightly better recall than Logistic Regression, the latter strikes a better balance between accuracy and simplicity, making it more practical for deployment in real-time applications.
- **Logistic Regression Is Deployment-Ready:** Its high performance, efficiency, and ease of integration into web applications make Logistic Regression the most suitable choice for this project's final implementation.
- **Room for Improvement:** Although the results are promising, further enhancements could be made by exploring deep learning models (like LSTMs or BERT) or incorporating additional features such as sentiment scores, named entities, or metadata.

V. IMPLEMENTATION

To demonstrate the practical application of the trained machine learning model, a web-based interface was developed using Streamlit, a Python framework designed for building fast and interactive data applications. This implementation bridges the gap between theoretical research and real-world usability by allowing end-users to interact with the model in real time.

5.1 Overview of the Web Application

The web app serves as an accessible tool that classifies news content as either "Real" or "Fake" based on the text entered by the user. It enables users—journalists, content reviewers, educators, and the general public—to verify the authenticity of a news article or headline quickly.

5.2 Technologies Used

Frontend and App Framework

- **Streamlit:** Used for designing the user interface and hosting the web app. Streamlit allows for rapid prototyping and seamless integration with Python-based ML models.

Backend and Model Integration

- **Python:** Core programming language used for model training, data preprocessing, and back-end logic.
- **Scikit-learn:** Employed for implementing the trained machine learning models (e.g., Logistic Regression).
- **NLTK:** Utilized for preprocessing text, including stopword removal and tokenization.
- **TF-IDF Vectorizer:** Used for transforming raw input text into numerical vectors suitable for classification.

5.3 Functional Features

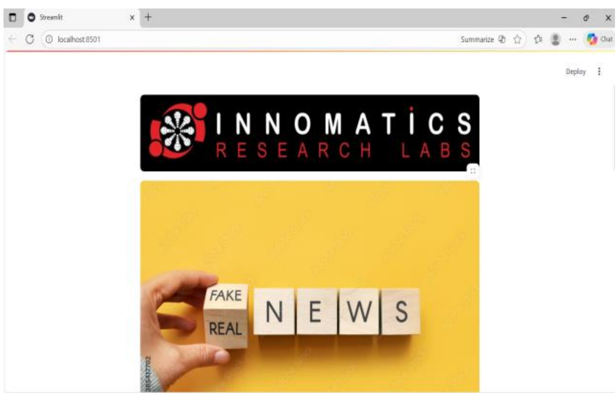
- **Text Input Field:** Users can enter either a news headline or a complete article into a text box provided in the app.
- **Prediction Button:** Upon clicking the "Predict" button, the input is preprocessed (lowercased, tokenized, and vectorized using the TF-IDF model) and passed through the trained machine learning classifier (e.g., Logistic Regression).
- **Output Display:** The app instantly returns a result—either "Fake News" or "Real News"—based on the model's prediction. The response is designed to be fast and readable.
- **Accuracy Indicator (Optional):** The app may display the confidence level or accuracy of the model, helping users understand the reliability of the prediction.

5.4 User Experience (UX)

The web application is designed with a clean and minimalistic interface, ensuring ease of use for non-technical users. It requires no installation or complex inputs, making it ideal for:

- **General Public:** As a verification tool during online browsing.
- **Fact-Checkers and Journalists:** To pre-screen articles during editorial review.
- **Educational Use:** For demonstrating machine learning applications in classrooms or workshops.

5.5 Deployment and Accessibility



- **Local Deployment:** The app can be run locally on any system with Python installed using a simple command: streamlit run app.py
- **Cloud Deployment (Optional):** For broader accessibility, the app can be hosted on cloud platforms such as:
- Streamlit Cloud
- Heroku
- AWS or Google Cloud Platform (GCP)

Deployment ensures that users can access the model via a web link without needing to install or configure anything.

5.6 Advantages of the Web-Based Interface

- Real-time feedback on news authenticity
- No programming knowledge required
- Lightweight and fast performance
- Easily extendable for future upgrades (e.g., deep learning models, multilingual support)

VI. CONCLUSION

The increasing prevalence of misinformation on the internet—especially on social media platforms—has made fake news detection an urgent and vital area of research. This study demonstrates the potential of machine learning techniques, combined with Natural Language Processing (NLP), to provide a reliable, automated solution to this growing challenge.

Among the models tested, Logistic Regression emerged as the most effective algorithm for binary classification of news articles, achieving an impressive accuracy of approximately 92%. This success can be attributed to the model's simplicity, robustness, and compatibility with high-dimensional, sparse data generated through TF-IDF vectorization.

The study's multi-phase methodology—including dataset collection, text preprocessing, model training, evaluation, and deployment—presents a complete end-to-end framework for fake news detection. It shows how machine learning can transition from academic theory to a practical, user-friendly solution capable of operating in real-time.
Furthermore, the implementation of a Streamlit-based web application demonstrates the applicability of the model in real-world scenarios.

This interface enables users—including media professionals, educators, and the general public—to input any headline or news article and receive instant feedback on its credibility. It provides a stepping stone toward democratizing access to fact-checking tools, which is particularly important in today's fast-paced digital landscape.

Key Takeaways:

- **Efficiency:** The system provides rapid, automated classification without human intervention.
- **Accuracy:** The selected ML models, especially Logistic Regression, yield high precision and recall scores, making them suitable for production use.

- **Accessibility:** The lightweight, browser-accessible app design ensures that the tool is usable by individuals with no technical background.
- **Impact:** By integrating this solution into digital platforms, it can play a significant role in curbing the spread of misinformation, promoting media literacy, and restoring trust in digital news content.

Future Work

While the current implementation has demonstrated promising results using classical machine learning techniques, fake news detection remains a complex and evolving challenge. As misinformation tactics become more sophisticated, so must the tools designed to counter them. This section outlines several important areas for future enhancement:

Incorporate Deep Learning Models (e.g., BERT)

Traditional machine learning models like Logistic Regression are effective but often struggle with capturing deep contextual meaning in text. Integrating advanced deep learning models, particularly transformer-based architectures like BERT (Bidirectional Encoder Representations from Transformers), can significantly improve performance by:

- Understanding the context and semantics of words in a sentence, rather than relying on isolated keywords.
- Capturing long-range dependencies, such as detecting subtle cues of misinformation embedded across paragraphs.
- Enabling transfer learning, where pre-trained models can be fine-tuned on specific fake news datasets for enhanced accuracy.

These models have already proven superior in many NLP tasks, including sentiment analysis, text classification, and question answering.

Extend Support to Multiple Languages

Most existing models, including the one developed in this project, are trained on English-language datasets, limiting their global applicability. However, fake news is not confined to one language—it spreads across social and regional boundaries. Future development should aim to:

- Train multilingual models or develop language-specific versions for widely spoken languages (e.g., Hindi, Spanish, Arabic).
- Use multilingual transformers like mBERT or XLM-R to build cross-lingual classifiers.
- Expand datasets to include labeled news content from diverse linguistic and cultural backgrounds.

This would make the system more inclusive and beneficial for users in non-English-speaking regions.

Analyze Fake News

Fake news is increasingly multimodal—it often includes manipulated images, deepfakes, or misleading videos alongside text to increase believability. Detecting such fake content requires:

- Computer Vision techniques to analyze visual data (e.g., using CNNs for image verification).
- Multimodal models that can process and correlate information across text and media.
- Integration with APIs or tools like Google Reverse Image Search or Video Frame Analysis for visual consistency checking.

Developing models that combine text and image analysis will create a more holistic and robust detection system.

Improve Handling of Nuanced Language (e.g., Sarcasm, Satire, Bias)

Detecting sarcasm, satire, hyperbole, and bias is one of the most difficult aspects of NLP. These linguistic styles often mask the true intent of the content and confuse standard models. Addressing this challenge involves:

- Incorporating sentiment and intent analysis to gauge emotional tone and implicit meaning.
- Using context-aware embeddings (like those generated by BERT or GPT) to better understand linguistic subtleties.
- Training the model with datasets that include examples of sarcastic, satirical, and biased news to improve recognition accuracy.

This would reduce false positives and negatives, especially in borderline cases where the content is misleading but not outright false.

Final Vision

By implementing these improvements, future versions of the system can evolve from a basic classifier to a comprehensive, intelligent misinformation detection platform—capable of real-time, multilingual, and multimodal analysis with a deeper understanding of language and intent.

REFERENCES

1. T. Mitra et al., "Combating Fake News: A Survey," IEEE Access, vol. 9, pp. 15692–15715, 2021.
2. D. Nguyen et al., "Fake News Detection Using Machine Learning," ACM Computing Surveys, vol. 53, no. 5, 2020.
3. K. Shu et al., "Fake News Detection on Social Media," IEEE Intelligent Systems, vol. 34, no. 6, 2019.
4. F. Pedregosa et al., "Scikit-learn: Machine Learning in Python", Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.
5. H. Allcott and M. Gentzkow, "Social media and fake news in the 2016 election," Journal of Economic Perspectives, vol. 31, no. 2, pp. 211–236, 2017.
6. S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false news online," Science, vol. 359, no. 6380, pp. 1146–1151, 2018.
7. A. Ahmed and L. Traore, "Detecting opinion spams and fake news using text classification," Procedia Computer Science, vol. 112, pp. 2291–2299, 2017.
8. Z. Zhang, D. Xu, and C. Yu, "Fake news detection with deep learning: A natural language processing perspective," Expert Systems with Applications, vol. 165, 2021.
9. R. K. Kaliyar, A. Goswami, and P. Narang, "DeepFakE: Improving fake news detection using tensor decomposition-based deep neural network," Neural Processing Letters, vol. 51, pp. 2407–2426, 2020.
10. A. Ruchansky, S. Seo, and Y. Liu, "CSI: A hybrid deep model for fake news detection," in Proc. CIKM, pp. 797–806, 2017.
11. M. Potthast et al., "A stylometric inquiry into hyperpartisan and fake news," in Proc. ACL, 2017.
12. B. Zhou and J. Zafarani, "Fake news: A survey of research, detection methods, and opportunities," ACM Computing Surveys, vol. 53, no. 5, pp. 1–40, 2020.
13. S. Wang, Y. Yao, and C. Li, "Beyond news contents: The role of social context for fake news detection," in Proc. WWW, 2018.
14. A. Kaliyar, A. Goswami, and P. Narang, "FakeBERT: Fake news detection in social media with a BERT-based deep learning approach," Multimedia Tools and Applications, vol. 80, no. 8, pp. 11765–11788, 2021.
15. J. Devlin et al., "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, 2019.
16. Goodfellow et al., "Generative adversarial nets," in Advances in Neural Information Processing Systems, 2014.
17. M. Ott et al., "A survey on early detection of fake news," Information Processing & Management, vol. 58, no. 6, 2021.
18. A. Alsmadi and T. O'Brien, "An ensemble learning approach for detecting fake news," Future Generation Computer Systems, vol. 101, pp. 389–403, 2019.
19. K. K. Horne and S. Adali, "This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news," Fake News Workshop at NAACL, 2017.
20. S. Reddy, "Fake news detection using machine learning algorithms," International Journal of Computer Applications, vol. 182, no. 40, pp. 22–26, 2018.
21. R. Baly et al., "Multitask learning for fine-grained fake news detection," Information Processing & Management, vol. 59, no. 2, 2022.
22. D. Rubin, N. Conroy, and Y. Chen, "Towards news verification: Deception detection methods for news discourse," Proceedings of the ACL, 2015.

23. A. Sinha et al., "Fake news detection: State-of-the-art and future research directions," Expert Systems with Applications, vol. 197, 2022.
24. K. Singh et al., "A review of machine learning-based approaches for fake news detection," Social Network Analysis and Mining, vol. 10, no. 1, 2020.
25. H. Jin, H. Cao, and Y. Guo, "Multimodal fusion with recurrent neural networks for rumor detection on microblogs," in Proc. MM, 2017.
26. P. Resnick et al., "Rumor Lens: A system for analyzing the impact of rumors and corrections in social media," ACM CSCW, 2014.
27. A. Agarwal and R. Alharbi, "The rise of fake news detection using machine learning," International Journal of Advanced Computer Science and Applications, vol. 10, no. 5, 2019.
28. NLTK Project, "Natural Language Toolkit Documentation," Available: https ://www.nltk.org/
29. Streamlit Team, "Streamlit: The fastest way to build and share data apps," Available: https: /streamlit.io/
30. A. Lazer et al., "The science of fake news," Science, vol. 359, no. 6380, pp. 1094–1096, 2018.