2026 Volume: 02 Issue: 06 www.ijrpa.com ISSN 2456-9995 Review Article

International Journal Research Publication Analysis

Page: 01-17

A HYBRID MACHINE LEARNING APPROACH FOR FAKE NEWS DETECTION USING NATURAL LANGUAGE PROCESSING TECHNIQUES

***Rakesh Maity, Partha Chakraborty, Subhadip Paul, Barun Mazumdar, Ashis De**

Department of Electronics and Communication Engineering, Gargi Memorial Institute of Technology, WB, India.

Article Received: 12 May 2026
Article Revised: 02 June 2026
Published on: 22 June 2026

***Corresponding Author: Rakesh Maity**
Department of Electronics and Communication Engineering, Gargi Memorial Institute of Technology, WB, India.

DOI: https://doi-doi.org/101555/ijrpa.7902

ABSTRACT

The exponential growth of digital media platforms and online information sources has significantly increased the spread of misinformation, commonly referred to as fake news. This widespread dissemination of misleading content poses serious challenges to societal stability, public trust, and informed decision-making. Fake news can influence political opinions, disrupt financial markets, and create public health risks, making its detection a critical problem in the modern information ecosystem. Traditional approaches such as manual fact-checking are insufficient due to the massive volume and high velocity at which information is generated and shared across digital platforms.

In response to these challenges, this work proposes a hybrid machine learning-based framework for automated fake news detection using Natural Language Processing (NLP) techniques. The proposed system is designed to analyze textual content and classify news articles as real or fake by leveraging both statistical and contextual features. The methodology incorporates multiple stages, including data preprocessing, feature extraction using Term Frequency–Inverse Document Frequency (TF-IDF), and classification using a combination of traditional machine learning algorithms such as Logistic Regression and deep learning models such as Long Short-Term Memory (LSTM) networks.

The hybrid approach aims to overcome the limitations of individual methods by combining the efficiency and simplicity of classical models with the advanced contextual understanding capabilities of deep learning architectures. While traditional models provide faster training

and lower computational requirements, deep learning models enable the system to capture sequential dependencies and semantic relationships within text. This integration is expected to improve classification accuracy and robustness across diverse datasets.

Although the system is currently in the design and development phase, the expected performance is based on insights from existing literature and prior studies. The proposed framework is anticipated to achieve high accuracy, precision, recall, and F1-score, making it suitable for practical deployment. Furthermore, the modular design of the system allows for future enhancements, including the integration of transformer-based models such as BERT and the incorporation of explainable artificial intelligence (XAI) techniques to improve model transparency and interpretability.

Overall, this research contributes toward the development of scalable and efficient fake news detection systems that can assist in combating misinformation in the digital age. The proposed hybrid framework provides a balanced approach that addresses both performance and computational constraints, making it a promising solution for real-world applications such as social media monitoring, automated news verification, and information filtering systems.

I. INTRODUCTION

The rapid advancement of internet technologies and the exponential growth of social media platforms have fundamentally transformed the way information is created, disseminated, and consumed. Platforms such as online news portals, blogs, and social networking sites have enabled users to access real-time information from anywhere in the world. However, this ease of access and sharing has also led to a significant rise in the spread of misinformation and fake news. Fake news refers to intentionally fabricated, misleading, or false information presented in the form of legitimate news with the objective of influencing public opinion, generating profit, or causing confusion.

The proliferation of fake news has become a major concern in modern society due to its far-reaching consequences. In the political domain, fake news has the potential to manipulate electoral outcomes and shape public perception of political figures and policies. During elections, misleading information can spread rapidly, influencing voters and undermining democratic processes. In the healthcare sector, the spread of false medical information—such as incorrect treatments or vaccine-related misinformation—can lead to serious public health risks. Similarly, in financial markets, fake news can cause fluctuations in stock prices and

economic instability by misleading investors. These examples highlight the critical need for effective mechanisms to detect and control the spread of fake news.

One of the primary challenges associated with fake news detection is the speed and scale at which information spreads across digital platforms. Unlike traditional media, where content is subject to editorial review, social media platforms allow anyone to publish and share information instantly. This results in a massive volume of data being generated continuously, making manual verification impractical. Although fact-checking organizations exist, their efforts are often reactive and cannot keep pace with the rapid dissemination of misinformation. Therefore, there is an urgent need for automated systems capable of analyzing and classifying news content in real time.

Machine learning and Natural Language Processing (NLP) have emerged as powerful tools for addressing this challenge. These techniques enable computers to process, analyze, and interpret human language, making it possible to identify patterns and features associated with fake news. Traditional machine learning approaches rely on statistical features such as word frequency, sentence structure, and stylistic patterns to classify text. While these methods are computationally efficient, they often fail to capture deeper contextual relationships and semantic meaning within the text.

In recent years, deep learning models such as Long Short-Term Memory (LSTM) networks have been introduced to overcome these limitations. LSTM models are capable of capturing sequential dependencies in textual data, allowing them to understand context and detect subtle linguistic cues that may indicate misinformation. However, deep learning models require large datasets and significant computational resources, which can limit their practical applicability in certain scenarios.

To address these challenges, this work proposes a hybrid machine learning approach that combines the strengths of traditional algorithms and deep learning models. The proposed system integrates efficient feature extraction techniques such as Term Frequency–Inverse Document Frequency (TF-IDF) with advanced models like LSTM to achieve a balance between computational efficiency and contextual understanding. By leveraging both statistical and semantic features, the system aims to improve classification accuracy and robustness.

The primary objective of this research is to design and develop a scalable and efficient fake news detection framework that can accurately classify news articles as real or fake based on their textual content. The proposed system is intended to serve as a foundation for future

enhancements, including the integration of transformer-based models and explainable AI techniques for improved transparency and performance.

II. Literature Review

The problem of fake news detection has attracted significant attention in recent years due to the increasing influence of digital media and the rapid spread of misinformation across online platforms. Researchers have explored various computational approaches to address this issue, ranging from traditional machine learning techniques to advanced deep learning and transformer-based models. This section presents a comprehensive review of existing methodologies, highlighting their strengths, limitations, and areas for improvement.

A. Traditional Machine Learning Approaches

Early research in fake news detection primarily relied on traditional machine learning algorithms such as Naive Bayes, Logistic Regression, Decision Trees, and Support Vector Machines (SVM). These approaches focus on extracting handcrafted features from textual data, including word frequency, n-grams, syntactic patterns, and stylistic features. Naive Bayes classifiers are widely used due to their simplicity and efficiency in handling large datasets. They operate under the assumption of feature independence and perform well in basic text classification tasks. Logistic Regression, another commonly used algorithm, provides probabilistic outputs and is effective for binary classification problems such as fake versus real news detection. Similarly, Support Vector Machines are known for their ability to handle high-dimensional feature spaces and are often used in text classification due to their robustness. Despite their advantages, traditional machine learning models have several limitations. They rely heavily on feature engineering, which requires domain knowledge and can be time-consuming. Moreover, these models struggle to capture semantic meaning and contextual relationships within text. For instance, they may fail to detect sarcasm, ambiguity, or subtle linguistic cues that are often present in fake news content. As a result, their performance is limited when dealing with complex and nuanced textual data.

B. Deep Learning-Based Approaches

With the advancement of deep learning, researchers have shifted towards models that can automatically learn feature representations from data. Deep learning approaches such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) networks, have shown significant improvements in fake news detection tasks. CNN-based models are effective in capturing local patterns and identifying key phrases or features within text. They apply convolutional

filters to extract important features and are often used for sentence-level classification. However, CNNs are limited in capturing long-range dependencies within text sequences. LSTM networks, on the other hand, are specifically designed to handle sequential data and can capture long-term dependencies. They use memory cells and gating mechanisms to retain relevant information over time, making them highly suitable for text analysis. LSTM-based models can understand context, detect subtle variations in language, and identify patterns indicative of fake news. Several studies have demonstrated that LSTM models outperform traditional machine learning methods in terms of accuracy and robustness. However, these models require large amounts of labeled data and significant computational resources for training. Additionally, deep learning models are often considered "black-box" systems, making it difficult to interpret their predictions.

C. Hybrid Approaches

To overcome the limitations of individual models, researchers have proposed hybrid approaches that combine traditional machine learning techniques with deep learning models. These approaches aim to leverage the strengths of both methodologies. For example, TF-IDF can be used for feature extraction to capture statistical importance, while LSTM models can be employed to capture contextual relationships within text. Logistic Regression can serve as a baseline classifier, while deep learning models can enhance performance by learning complex patterns. Hybrid models offer a balance between computational efficiency and predictive performance. They reduce reliance on extensive feature engineering while still benefiting from the contextual understanding provided by deep learning. As a result, hybrid approaches have gained popularity in recent fake news detection research.

D. Transformer-Based Models

Recent advancements in Natural Language Processing have introduced transformer-based architectures such as BERT (Bidirectional Encoder Representations from Transformers), GPT, and RoBERTa. These models have achieved state-of-the-art performance in various NLP tasks, including text classification, sentiment analysis, and fake news detection. Transformer models utilize attention mechanisms to capture relationships between words in a sentence, regardless of their position. This allows them to understand context more effectively compared to traditional and sequential models. BERT, for example, processes text bidirectionally, enabling it to capture both left and right context simultaneously. Despite their superior performance, transformer-based models have certain drawbacks. They require high computational power, large memory capacity, and extensive training data. This makes them

less suitable for real-time applications or deployment inresource-constrained environments. Additionally, fine-tuning these models can be complex and time-consuming.

E. Research Gap and Motivation

Although significant progress has been made in fake news detection, several challenges remain. Traditional machine learning models are efficient but lack contextual understanding, while deep learning and transformer-based models offer improved performance at the cost of increased computational complexity and reduced interpretability.

Furthermore, many existing approaches focus solely on improving accuracy without considering scalability and practical deployment. There is a need for systems that can balance performance, efficiency, and adaptability while being suitable for real-world applications.

This work aims to address these challenges by proposing a hybrid machine learning framework that combines the strengths of traditional and deep learning approaches. By integrating TF-IDF-based feature extraction with LSTM-based contextual learning, the proposed system seeks to achieve improved accuracy while maintaining computational efficiency.

III. Proposed Methodology

The proposed system is designed as a hybrid machine learning framework that integrates Natural Language Processing (NLP) techniques with both traditional and deep learning models to accurately classify news articles as real or fake. The methodology follows a structured pipeline consisting of multiple stages, each responsible for transforming raw textual data into meaningful representations suitable for classification.

The overall workflow of the system can be described as:

Input Text → Preprocessing → Feature Extraction → Model Training → Prediction Output

Each component of the system is described in detail below.

A. System Overview and Workflow

The system begins by accepting raw news text as input, which may include articles, headlines, or social media posts. This raw data is typically unstructured and contains noise such as punctuation, redundant words, and inconsistent formatting. Therefore, it undergoes a sequence of preprocessing steps to ensure uniformity and improve data quality.

After preprocessing, the cleaned text is transformed into numerical form using feature extraction techniques such as TF-IDF. These numerical representations are then fed into

machine learning models for training and classification. The final output of the system is a binary label indicating whether the news is real or fake.

This modular pipeline ensures flexibility, allowing individual components to be improved or replaced without affecting the entire system.

B. Data Preprocessing

Data preprocessing is a critical step in the proposed system, as the quality of input data directly influences model performance. Raw textual data often contains irrelevant and inconsistent elements that can negatively impact classification accuracy.

The preprocessing pipeline includes the following steps:

1. Tokenization

Tokenization involves splitting the input text into smaller units called tokens, typically words or subwords. This step enables the system to analyze text at a granular level.

2. Lowercasing

All characters in the text are converted to lowercase to ensure uniformity. This prevents duplication of words due to case differences (e.g., “News” and “news”).

3. Stopword Removal

Commonly used words such as “the”, “is”, “and”, which do not contribute meaningful information to classification, are removed. This reduces dimensionality and improves efficiency.

4. Lemmatization

Lemmatization reduces words to their base or root form. For example, “running”, “runs”, and “ran” are converted to “run”. This ensures consistency in word representation.

5. Noise Removal

Special characters, punctuation marks, URLs, and numerical values are removed to clean the dataset.

These preprocessing steps collectively enhance the quality of the dataset and ensure that only relevant textual information is retained for further analysis.

C. Feature Extraction

Once preprocessing is completed, the textual data must be converted into numerical form so that machine learning models can process it. This is achieved using the Term Frequency–Inverse Document Frequency (TF-IDF) technique.

TF-IDF Representation

TF-IDF measures the importance of a word in a document relative to a collection of documents (corpus). It is calculated as:

$$TF - IDF = TF \times \log(\frac{N}{DF})$$

Where:

TF (Term Frequency) represents how frequently a word appears in a document

DF (Document Frequency) represents the number of documents containing the word

N is the total number of documents

TF-IDF assigns higher weights to words that are unique and informative, while reducing the importance of commonly occurring words. This helps in identifying key features that distinguish fake news from real news.

D. Model Architecture

The proposed system uses a hybrid architecture that combines traditional machine learning with deep learning models to leverage the strengths of both approaches.

1. Logistic Regression (Baseline Model)

Logistic Regression is used as a baseline classifier due to its simplicity and efficiency. It works well with TF-IDF features and provides fast training and prediction. It serves as a reference point for evaluating the performance of more complex models.

2. BiLSTM (Deep Learning Model)

The Long Short-Term Memory (LSTM) network is used to capture sequential dependencies in text. In this work, a Bidirectional LSTM (BiLSTM) is employed, which processes text in both forward and backward directions, allowing the model to understand context more effectively.

Why BiLSTM?

Captures past and future context simultaneously

Handles long-term dependencies in text

Improves understanding of sentence structure

3. Attention Mechanism

An attention layer is added after the BiLSTM layer to enhance model performance. The attention mechanism assigns different weights to words in a sentence based on their importance.

For example, in fake news detection:

Words like “shocking”, “breaking”, “exclusive” may carry higher importance

Attention helps the model focus on such keywords

This improves interpretability and accuracy.

4. Dense Layer and Output

The final dense (fully connected) layer processes the extracted features and produces the output. A sigmoid activation function is used to generate a probability value between 0 and 1.

Output = 0 → Real News

Output = 1 → Fake News

E. Working of LSTM (Detailed Explanation)

LSTM networks are designed to overcome the limitations of traditional Recurrent Neural Networks (RNNs), particularly the problem of vanishing gradients. They achieve this through a memory cell and three types of gates:

1. Input Gate

Controls how much new information is added to the memory.

2. Forget Gate

Determines which information should be removed from the memory.

3. Output Gate

Controls how much information from the memory is used for prediction.

These gates allow the LSTM to retain important information over long sequences, making it highly effective for text-based tasks such as fake news detection.

F. Training Strategy

The model is trained using labeled datasets consisting of real and fake news articles. The dataset is typically divided into:

Training set (80%)

Testing set (20%)

During training, the model learns patterns associated with fake and real news. Optimization is performed using gradient descent algorithms such as Adam optimizer.

G. Mathematical Formulation

The model is trained using the Binary Cross-Entropy Loss Function, defined as:

$$L = -\frac{1}{N}\sum(y log(\hat{y}) + (1-y) log(1-\hat{y}))$$

Where:

y is the actual label

$\hat{y}$ is the predicted probability

N is the number of samples

This loss function measures the difference between predicted and actual values and guides the optimization process.

H. Justification of Hybrid Approach

The hybrid approach is chosen to overcome the limitations of individual models:

Traditional models (Logistic Regression) provide efficiency but lack contextual understanding

Deep learning models (LSTM) provide context but require high computation

By combining both:

The system achieves better accuracy

Maintains computational efficiency

Improves generalization across datasets

This makes the proposed framework more practical for real-world deployment.

IV. Implementation Plan

The implementation of the proposed fake news detection system will be carried out in a structured and systematic manner using widely adopted tools and technologies in the field of machine learning and Natural Language Processing (NLP). Although the system is currently in the design phase, this section outlines a comprehensive plan for developing, training, and evaluating the proposed hybrid model.

A. Development Environment and Tools

The implementation will be performed using the Python programming language, which is widely used for machine learning and NLP applications due to its extensive library support and ease of use. The following tools and libraries will be utilized:

NumPy and Pandas: For data manipulation and preprocessing

NLTK / spaCy: For text preprocessing tasks such as tokenization, stopword removal, and lemmatization

Scikit-learn: For implementing traditional machine learning models such as Logistic Regression and TF-IDF vectorization

Tensor Flow / Keras or PyTorch: For building and training deep learning models such as LSTM

Matplotlib / Seaborn: For data visualization and performance analysis

The development environment may include platforms such as Jupyter Notebook, Google Colab, or any integrated development environment (IDE) like PyCharm or VS Code.

B. Dataset Selection and Description

The system will be trained and evaluated using a publicly available dataset, such as the Fake and Real News Dataset from Kaggle. This dataset typically contains labeled news articles categorized as fake or real.

Dataset Characteristics:

Textual content (news articles or headlines)

Binary labels (Fake = 1, Real = 0)

Balanced or semi-balanced class distribution

Before training, the dataset will be examined for:

Missing values

Duplicate entries

Class imbalance

If necessary, techniques such as data balancing or resampling may be applied to ensure fair model training.

C. Data Preprocessing Pipeline

The preprocessing pipeline transforms raw text into clean and structured data suitable for model input. The following steps will be implemented:

Loading the Dataset

The dataset will be imported using Pandas and converted into a structured format such as a DataFrame.

Text Cleaning

Removal of punctuation, special characters, URLs, and irrelevant symbols to reduce noise.

Tokenization

Splitting text into individual words or tokens for analysis.

Stopword Removal

Eliminating common words that do not contribute to classification.

Lemmatization

Converting words to their root forms to ensure consistency.

Lowercasing

Standardizing all text to lowercase to avoid duplication.

These steps ensure that the data is clean, consistent, and ready for feature extraction.

D. Feature Extraction Implementation

Feature extraction will be performed using the TF-IDF Vectorizer from Scikit-learn.

Steps:

Convert preprocessed text into numerical vectors

Limit vocabulary size to reduce dimensionality (optional)

Use n-grams (unigrams/bigrams) to capture word sequences

The resulting feature matrix represents each document numerically and can be directly used for model training.

E. Model Development

1. Logistic Regression Model

The Logistic Regression model will be implemented as a baseline classifier.

Steps:

Train model using TF-IDF features

Optimize hyperparameters such as regularization strength

Evaluate performance using test data

This model provides a reference for comparing performance with deep learning models.

2. LSTM Model Implementation

The LSTM model will be developed using TensorFlow or PyTorch.

Steps:

Convert text into sequences using tokenization

Apply padding to ensure uniform sequence length

Use embedding layer to convert words into dense vectors

Add LSTM or BiLSTM layers for sequence learning

Add dense layer with sigmoid activation for classification

Hyperparameters such as number of layers, hidden units, batch size, and epochs will be tuned to achieve optimal performance.

F. Training Strategy

The dataset will be split into training and testing sets, typically using an 80:20 ratio.

Training Process:

Train models on training data

Validate performance using validation set (optional)

Use optimizer such as Adam for gradient descent

Monitor loss and accuracy during training

To prevent overfitting:

Dropout layers may be added

Early stopping may be applied

G. Model Evaluation

After training, the models will be evaluated using standard performance metrics:

Accuracy

Precision

Recall

F1-score

Additionally, a confusion matrix will be used to analyze classification performance in detail.

H. Comparison and Analysis

The performance of different models will be compared using tabular representation.

Example:

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**
Logistic Regression	Moderate	Moderate	Moderate	Moderate
LSTM	High	High	High	High
Hybrid Model	Highest	High	High	High

This comparison helps identify the strengths and weaknesses of each approach.

I. Deployment Considerations (Optional Extension)

Although not part of the current implementation phase, the system can be extended for real-world deployment:

Integration with web applications

Real-time fake news detection APIs

Social media monitoring tools

J. Challenges and Risk Mitigation

Several challenges may arise during implementation:

Data Imbalance → Use resampling techniques

Overfitting → Apply regularization and dropout

High Computational Cost → Optimize model architecture

Proper planning ensures that these challenges are effectively managed.

V. Expected Results and Analysis

The proposed hybrid fake news detection system is designed to improve classification performance by combining the strengths of traditional machine learning and deep learning techniques. Since the system is currently in the design and implementation phase, the results presented in this section are based on expected outcomes derived from prior research studies and theoretical analysis.

A. Performance Expectations

The proposed hybrid model is expected to achieve high classification accuracy by effectively leveraging both statistical and contextual features. Based on findings from existing literature, machine learning models such as Logistic Regression typically achieve accuracy in the range of 85%–92%, while deep learning models such as LSTM can achieve higher accuracy due to their ability to capture contextual dependencies.

By combining these approaches, the hybrid model is expected to achieve:

Accuracy: Approximately 90%–97%

Precision: High precision in identifying fake news articles

Recall: Improved ability to detect actual fake news cases

F1-score: Balanced performance between precision and recall

These performance metrics indicate that the system is capable of reliably distinguishing between real and fake news content.

B. Comparative Analysis of Models

To evaluate the effectiveness of the proposed approach, the performance of different models will be compared. The following table illustrates the expected comparison:

Model	Accuracy	Precision	Recall	F1-score
Logistic Regression	Moderate	Moderate	Moderate	Moderate
LSTM	High	High	High	High
Hybrid Model	Highest	High	High	High

Analysis:

Logistic Regression:

Provides fast and efficient classification but lacks contextual understanding, leading to moderate performance.

LSTM Model:

Captures sequential dependencies and contextual relationships, resulting in improved accuracy compared to traditional models.

Hybrid Model:

Combines the advantages of both approaches, leading to superior performance in terms of accuracy and robustness.

C. Confusion Matrix Analysis

A confusion matrix is used to evaluate the performance of the classification model by comparing predicted and actual values.

The confusion matrix consists of four components:

True Positive (TP): Fake news correctly identified as fake

True Negative (TN): Real news correctly identified as real

False Positive (FP): Real news incorrectly classified as fake

False Negative (FN): Fake news incorrectly classified as real

Expected Observations:

High TP and TN values indicate strong model performance

Low FP ensures that real news is not wrongly flagged as fake

Low FN ensures that fake news is effectively detected

The hybrid model is expected to minimize both FP and FN, thereby improving overall reliability.

D. Graphical Analysis (Expected)

Although the system is not yet implemented, the following graphical representations are expected to be used for performance analysis:

1. Accuracy Comparison Graph

A bar chart comparing accuracy across different models (Logistic Regression, LSTM, Hybrid).

2. Loss vs Epoch Curve

For the LSTM model, a graph showing training and validation loss over epochs is expected. A decreasing trend indicates effective learning.

3. Precision-Recall Curve

This graph illustrates the trade-off between precision and recall, helping evaluate model performance under different thresholds.

4. Confusion Matrix Visualization

A heatmap representation of the confusion matrix provides a clear understanding of classification results.

E. Key Observations and Insights

Based on theoretical analysis and previous studies, the following observations are expected:

The hybrid model will outperform individual models due to its ability to capture both statistical and contextual features

TF-IDF improves feature representation by emphasizing important words

LSTM enhances contextual understanding, especially for long text sequences

Attention mechanisms (if applied) further improve performance by focusing on relevant words

F. Limitations of Expected Results

While the proposed system is expected to perform well, certain limitations must be considered:

Performance depends on dataset quality and size

Deep learning models may require high computational resources

Model performance may vary across different domains and datasets

G. Practical Implications

The expected results indicate that the proposed system can be effectively used in real-world applications such as:

Automated fake news detection systems

Social media content moderation

News verification tools

Information filtering systems

VI. CONCLUSION

In this work, a hybrid machine learning framework for fake news detection has been proposed using Natural Language Processing (NLP) techniques. The system is designed to address the growing challenge of misinformation in digital media by combining traditional machine learning algorithms with deep learning models. By integrating feature extraction

methods such as TF-IDF with models like Logistic Regression and LSTM, the proposed approach aims to achieve a balance between computational efficiency and contextual understanding.

The study highlights the importance of preprocessing techniques and feature engineering in improving model performance. Additionally, the use of deep learning models enables the system to capture semantic relationships and sequential dependencies within textual data, which are crucial for identifying subtle patterns in fake news content. The hybrid approach is expected to enhance classification accuracy and provide more robust results compared to individual models.

Although the system is currently in the design and development phase, the proposed methodology and implementation plan provide a solid foundation for future development. The expected results indicate that the system can achieve reliable performance and can be extended to handle large-scale datasets and real-time applications.

VII. REFERENCES

1. Devlin et al., BERT Paper
2. Zhou & Zafarani, Fake News Survey
3. Kaggle Dataset
4. WWW.Wikipedia.org