Journal of Information Systems Engineering and Management
2025, 10(7s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Automatic Fake News Identification: A Hybrid CNN-LSTM-Logistic Regression Approach

Pummy Dhiman1, *, Amandeep Kaur1
1Chitkara University Institute of Engineering and Technology. Chitkara University, Punjab, 140401, India
pummy.dhiman@chitkara.edu.in, amandeep@chitkara.edu.in

ARTICLE INFO

Received: 10 Oct 2024

Revised: 11 Dec 2024

Accepted: 24 Dec 2024

ABSTRACT

The news provides important insights into current events and acts as a vital window into the world. The propagation of fake news, poses a serious problem. News that seems to present genuine but is made up is considered fake news. Such fake news can propagate inadvertently or on purpose, foment strife, and erode trust. Identifying fake news has been the focus of various studies to address this issue. To contribute in this direction, we proposed a stacking approach that combines convolutional neural networks (CNN) and long short-term memory (LSTM). We use logistic regression (LR) as a metaclassifier for final classification. We used accuracy, precision, recall, and F1-score as performance evaluation metrics on a real-world dataset. The dataset included in this study reflects a wide range of information and consists of both content from social media platforms and news items from reliable sources. We use McNemar's test to determine the statistical significance of the model's performance. The proposed hybrid approach yields impressive results: 95.19% accuracy, 95.05% precision, 95.54% recall, and 95.29% F1-score. These findings highlight the hybrid model's efficacy in correctly identifying fake news, supporting social peace and the preservation of real news.

Keywords: Deep learning, Identification, Fake news, Hybrid model, Stacking, Internet access, social media, Technology.

INTRODUCTION

In the realm of the digital landscape, people are now moving from traditional media to modern media to get news[1]. With the affordable rate of mobile devices and internet packs, it is easy to get any information with just a single click. Now the Internet has become the fourth basic requirement of human life—food, clothing, and shelter. Even social media has made it very easy to get connected with our loved ones, despite the distance. Social media is generating massive amounts of data, and not everything that is presented there is authentic. The generated content may be a fake news. It is news that pretends to be true but is actually fake. In the digital space, there is fake news along with real news. Here, anyone can become an influencer and share their thoughts and any news. But it also leads to the spread of information pollution. By information pollution, we mean misinformation, disinformation, and mal-information. In 2019, there was a surge in fake news-related events in India like the CAA, NRC, Article 370, and Pulwama attacks, which was termed the "Year of Fake News"[2]. It is not that fake news is only propagated on social media; news outlets also contribute to it. Fake news is a danger to humankind. It also disturbs the country's peace and economy. There are various steps taken by the government to combat the menace of fake news. The Information Technology (Intermediary Guidelines and Digital Media Ethics Code) Regulations, 2021, which take effect on February 25, 2021, require social media platforms to remove content within 36 hours of obtaining a court order or official directive. In order to cooperate with law enforcement, the regulations also require these platforms to designate compliance officers and nodal officers. Although the Indian Penal Code (IPC) does not contain any explicit provisions that target fake news, many sections of it can be used to penalize those responsible for spreading false information. Former Zimbabwean cricketer Heath Streak debunked death rumours that circulated in August 2023. He also demanded an apology from the source responsible for disseminating the false information, emphasizing the importance of verification and accountability in the era of social media. While talking about the Digital Personal Data Protection Act, 2023, the Indian Minister of State for IT and Electronics, Rajeev Chandrasekhar, emphasized the considerable difference in the speed and reach of misinformation and the truth. While highlighting that bogus news spreads considerably faster and reaches a much broader audience than real news.

Geoffrey Hinton, regarded as the "Godfather of AI," is concerned about the misinformation issue that generative AI-based technology could cause. Hinton underlined that the internet will be swamped with fake photos, videos, and texts and that ordinary people may no longer be able to tell what is true[3]. Because of technological breakthroughs, digital media can now be manipulated in ways that no one could have imagined twenty years ago[4]. Due to the ever-increasing volume of daily news, it is very tedious to manually develop a reliable method for identifying false news. As stated earlier, AI could result in fake news generation, but it can also be a technique for spotting fake news by identifying if there are any unmistakable linguistic patterns. Many studies have been conducted to counter this fake news dissemination issue. Since this is a global issue, numerous researchers from all over the world have created numerous machine learning[5], deep learning[6] and natural language processing[7] techniques for reliable fake news detection[8]. As the news is text-based, text corpora play a crucial role in the development and training of fake news detection systems; therefore, researchers also worked on the creation of developing data corpora to tackle fake news detection[9].

With the huge amount of data samples, deep learning techniques, including RNN and even one-dimensional CNN models, have proven effective in fake news identification and classification. The CNN and LSTM models' combined techniques have shown great results[10]. These techniques automatically extract features and handle data dependencies to find sequential patterns in text and make decisions about whether a news story is real or fake. In addition to the methodology, the input data is crucial to a model's performance since it determines how accurate the model will be. In this instance, the proposed hybrid approach[11]—which combines CNN, RNN, and MLP—demonstrated an excellent combination of cutting-edge methods while still managing to detect and categorize false news with an accuracy of 44.87%. The success of a model depends not only on the approach utilized but also on the input data as well as the pre-processing steps, which play a key role in gaining better accuracy. In this paper, a fake news classification method with multiple supervised ML techniques based on content and context level is illustrated and implemented.

The following are the key contributions of current study:

- It presents a hybrid approach that combines features extracted from both machine learning and deep learning technique to reap their benefits for fake news classification.
- A model is developed based on CNNs and LSTM. The results from these two models are then combined and fed into Logistic Regression for the final classification. This approach leverages both content and context-based features.
- This approach yields superior performance compared to traditional methods, offering a practical solution to combat bogus news and promoting information accuracy.

The outline of the paper looks like this: Section 2 provides a literature review. The mathematical definition of false news detection is presented in Section 3. Section 4 covers the research methodology, including the datasets, pre-processing, and vectorization that were applied. Section 5 presents the ML models and evaluation criteria used in this study. Section 6 represents the proposed hybrid method. Section 7 includes the comparative analysis, and Section 8 concludes with future directions.

1. Literature Review

This part represents some of the research conducted by various researchers to identify bogus news. NLP lets computers understand human language. It helps identify bogus news by analysing text content, linguistic trends, and sentiment. This experiment conducted by[12] tested Natural Language Processing (NLP) models' ability to anticipate tweet claims' fact-checkability. It covers COVID-19[13] and politics in Arabic, Bulgarian, English, Spanish, and Turkish. Each tweet was scored between 0 and 1, with higher ratings suggesting check-worthiness. Features plays a key role in detecting false news. It can be based on content level, context level, network level. A study conducted by authors [14] portrayed a wide variety of textual features in context of phony news. In order to facilitate the detection of fabricated news, the authors experimented with KNN, NB, RF, XGB, and SVM using a new set of features. The result showed XGB outperformed with 86% accuracy. The enormous amount of news generated day to day imposes a limit on ML; therefore, DL is used to handle the massive amount of content. CNN is a well-known DL-based model that is utilized in the text classification process[15]. Using an early fusion approach, the authors[16], classified fake news on the Fakeddit dataset for unimodality as well as multimodality. Accuracy rates of 77% for text detection using BERT, 75% for picture detection using ResNet50, and 85% for multimodality (text and image) combining BERT and ResNet50 were reported. Inspired by the transformer approach, authors proposed FND-NS[17], a binary classification model employing an effective weak supervision labelling scheme, for early fake news detection. 74.8% accuracy, and 74.9% F1 score obtained, all of which are higher than the baseline. This experiment used the English datasets NELA-GT-2019 and Fakedit. The proposed study in [18] suggests using an ensemble classifier to outperform individual-based classifiers. The technique also balances uneven data sets to reduce detection algorithm influence and improve accuracy. Experimental results on the PHEME demonstrated that the new features improved classification efficiency and accuracy when compared to numerous comparable rumour classification works using the same data. The data set was optimized using three ML models: Ensemble Voting Classifiers, Naïve Bayes, SVM, and KNN and results 78.54% accuracy. Training and testing the proposed models used WEKA. This work extracts social context features from tweet temporal features, propagation features, and user behaviour to account for user connections and rumour spreading. To identify bogus news by focusing on the fake Facebook users, authors of a study [19] presented two different classification methods: one using logistic regression, and the other based on Boolean crowdsourcing algorithms. This study offers a unique perspective that must be taken into account when identifying scam that is circulating online by focusing on social media, mainly Facebook likes made by genuine and fake users. Using the LIAR dataset, a DL based ensemble model [20] is proposed. Authors of the study[21] presented OPCNN-FAKE, an optimized CNN model architecture incorporating grid search for optimization, and n-grams with TF-IDF were used to represent features using ML and DL. This method fared better than competing models across four separate datasets. In the future, efforts may expand beyond the study of English to include the study of other languages and the incorporation of visual data. The stock market is just one more area that can be negatively impacted by fake news. It is undeniable that fake financial news has the power to sway public opinion and distort markets. A hybrid CNN-LSTM based model is proposed by authors in [10] to detect financial-related fake news. With a 92.1% accuracy, this method clearly excels. In order to detect and multi-class classification, researchers [11] developed an ensemble-based architecture based on CNN-BiLSTM and Multi-layer Perceptron (MLP).

By inspiring the hybrid approach of CNN-RNN, this current study deploys a strong feature extraction approach that combines CNN for capturing local textual patterns, LSTM for modelling sequential dependencies, and logistic regression for the final fake news classification. This combined architecture is to yield reliable outcomes, which will aid in the ongoing fight against disinformation across disciplines.

This study primarily addresses the research question: How can we make use of ML techniques along with the news content and context to enhance the accuracy of the fake news identification method? To deal with this issue, this article presents a hybrid model that utilizes both DL and ML techniques to distinguish between fabricated and genuine news stories.

2. Problem Definition

This section offers the formal mathematical definition of fake news detection. The aim of detecting false news is to analyse the credibility of news articles from various sources, including news channels and social media. In the context of detecting fake news, we have a collection of news articles and their respective label. The goal is to learn a prediction function, F, that takes a given news article 'a' and determines whether it is false news (F(a) = 1) or not (F(a) = 0).

This problem can be mathematically formulated as follows:

Given a dataset of news articles A and their associated binary labels Y, where:

• $A = \{a_1, a_2, ..., a_n\}$ represents the set of news articles, with 'n' representing the total number of news articles comprises the dataset.

• $L = \{l_1, l_2, ..., l_n\}$ represents news articles labels, with each label $l_n \rightarrow \{0, 1\}$, where

- 1 implies that the news is fake
- 0 implies that the news is real

For each news article 'a', we can extract features that can be interpreted as the feature vector 'X(a)'.

The goal is to derive a prediction function F that takes the feature vector 'X(a)' of a news and projected the news label, i.e., $F(a) \rightarrow \{0, 1\}$,

where:

- F(a) = 1 if news 'a' is projected as fake news
- F(a) = 0 if news 'a' is projected as real news

3. Methodology

This research makes use of Google Colab and a number of Python modules, including NLTK, an all-encompassing toolkit for natural language processing tasks like tokenization and lemmatization. We handle visualization with Matplotlib and Seaborn, and implement ML with Scikit-learn. These library packages collaborated to quicken both the running of the experiment and the analysis of the results[22]. Fig. 1 shows the methodology used in this investigation.

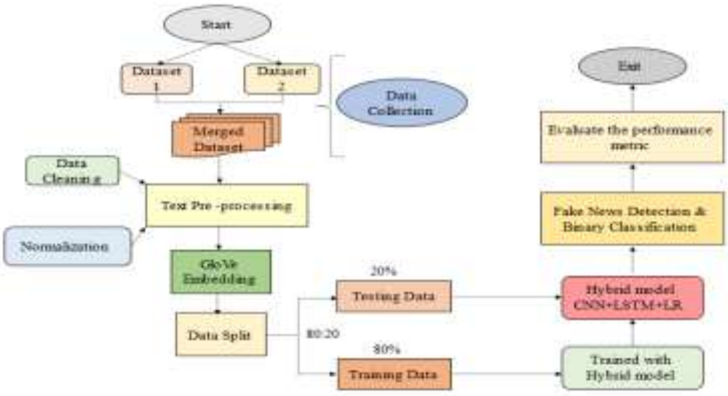

Fig. 1 Fake news identification process

The methodology of this study includes the following steps: (i) data collection and pre-processing; (iii) developing a stacked fake news detection approach; and (iv) measuring the model's effectiveness.

3.1. Dataset

The dataset used in current study comprises news articles from 2013-2021 including from various news outlets as well from social media. Table I represents the statistics of datasets.

Table I Dataset Statistic

Dataset	Fake News	Real News	Features
IFND (2021) [23]	19059	37809	Containing Indian news from mainstream news channels of heterogenous topics
FakeNewsIndia (2022) [24]	4803	-	Fake news incidents propagated on social media

3.2. Data Pre-processing

Before the data is pre-processed, it is analysed to determine what information is most crucial. This makes exploratory data analysis a crucial part of the natural language processing (NLP) effort. In Fig. 2, a graphic representation of the web sources of various media outlets is depicted. This demonstrates that Tribune India is the primary source for the news articles used in this analysis[23].

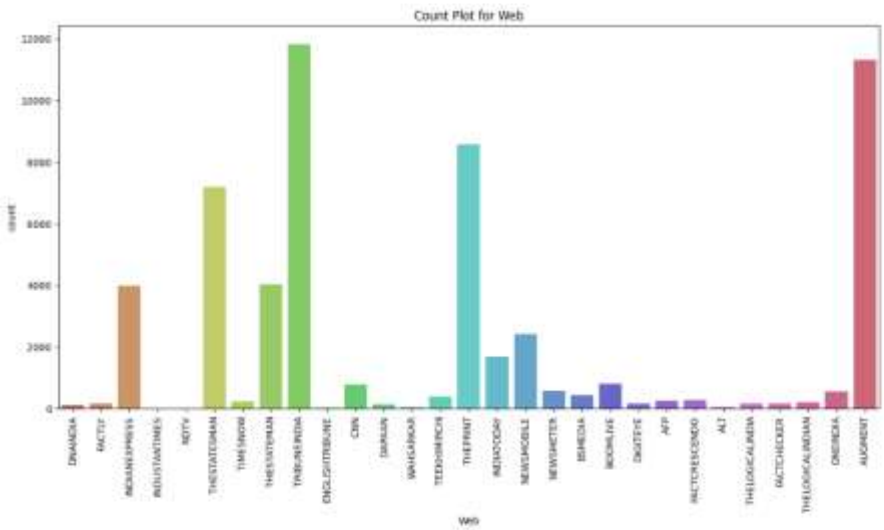

Fig. 2 Graphical representation of the web sources

News articles in dataset 1 cover a wide variety of topics beyond politics. There are a variety of domains and themes covered in this dataset, indicating that it will be suitable for extensive text analysis and topic modelling across a wide range of domains and topics such as Covid-19 [25], politics, violence[26] and many others.

The news instances from the dataset 2 are from social media [27] used in this study. This combination of data sources allows to investigate various modes of information dissemination on the web[24].

The two datasets, each of which is contained within its own data frame, are each represented by Pandas. The data frames from the two different data corpora were combined using the concatenation function that was included in this package. This merged corpus (Table II) will be used for preliminary data processing.

Table II Merged dataset statistic

Label	Count
Real News	37800
Fake News	23564

As shown in Fig. 3, there are many different pre-processing techniques used in this study.

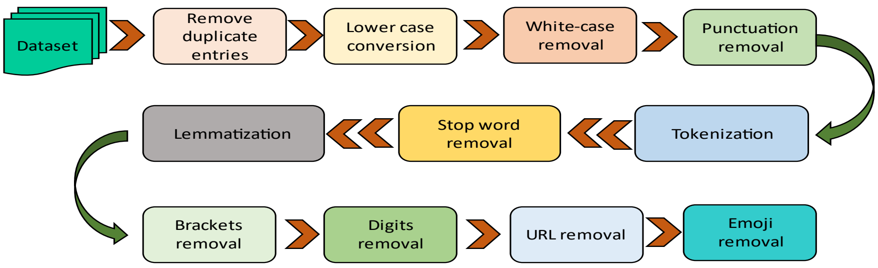

Fig. 3 NLP Pipeline

All uppercase letters are folded into lowercase ones to level the text. All punctuation, special symbols, and URLs are removed because they don't change the message. Stop words like “a”, “an”, “the”, “is”, “of” are eliminated from the text because they appear more often and provide less useful information. Remove square bracket text and number words from text. Emojis were important for expressing emotions, but not for fake news detection task. We also eliminated them. NLP uses en_US.utf-8 for American English text to ensure American English-compliant tokenization. We use NLTK lemmatization in DataFrame 'df_new' and its 'Statement' column for normalization. The Porter Stemmer reduces word affixes to root forms, whereas the WordNet Lemmatizer generates valid words from linguistic context. Both methods simplify word variants to improve computational efficiency and linguistic accuracy in text analysis.
Word clouds are a popular and elegant NLP text presentation method. The frequent used words in a corpus of text can be easily determined via data visualization. Fig. 4 shows the merged, pre-processed dataset word cloud.

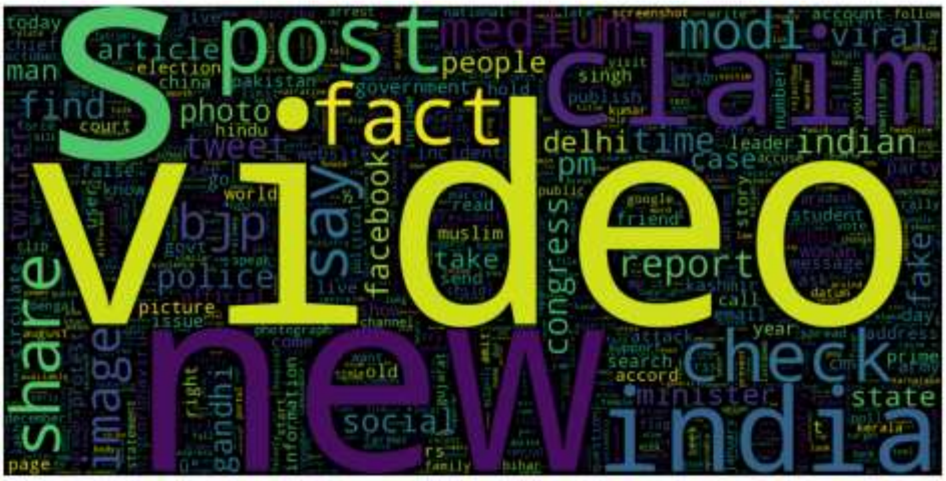

Fig. 4 Word-cloud of pre-processed dataset

As depicted by the figure, the pre-processed word cloud signifies that the text has undergone cleaning, resulting in the removal of any irrelevant content. After removing any special characters, the text in the word cloud has been effectively cleaned and is now prepared for input into machine learning classifiers.

3.3. Vectorization and Feature Extraction

Human language is unstructured, and ML models can't work directly on it. Therefore, it must be converted into numerical vectors, like matrix form. The process of converting textual data into numeric vectors is the fundamental concept of NLP, known as vectorization. It allows you to represent words, phrases, or documents as numerical values, making it possible to perform various NLP tasks. Vectorization is a subset of feature extraction. Feature extraction and vectorization are both ways to get text data ready for an ML model[28]. The goal of feature extraction is to get useful information from text and turn it into a set of features that can be used to assist ML models in distinguishing between authentic news and counterfeit ones. Here are some common methods for feature extraction in NLP:

- **Term Frequency-Inverse Document Frequency (TF-IDF):** Measures the importance of words in a document relative to their frequency across the entire dataset. It helps identify significant words specific to a document. It quantifies the importance of each term in a document relative to the entire corpus. Rarely used words are given more weight and captured by TF. Frequently used words have less weight and are captured by the IDF. TF is used for every sentence, and IDF is used for every word[29].

$$\text{TF-IDF}= \frac{No\ of\ repeatation\ of\ words\ in\ sentence}{No\ of\ words\ in\ sentence} * log_e\left(\frac{No\ of\ sentences}{No\ of\ sentences\ containing\ the\ word}\right)$$

In this way, the entire document is converted into numerical vectors and ready to be processed by ML models.

- **Word Embeddings:** Word embedding, also known as distributed word representation, is a method of representing words as numerical vectors, with words of related meaning clustered together. Word embedding makes this possible by recording a word's semantic and syntactic features in a massive corpus[29].

- **Global Vectors for Word Representation (GloVe):** It is used to analyse the word co-occurrence statistics within a corpus [30]. The tool is quite good at capturing both the global context and the semantic links between concepts. This particular word embedding technique is also commonly used and well-known. Using matrix factorization, it creates a vector representation of each word based on its frequency of occurrence in the given corpus. For many languages and domains, pretrained GloVe embeddings are available, eliminating the need to train embeddings from scratch and saving significant time and resources in the process. GloVe 100D word embeddings named 'glove.6B.100d.txt' are downloaded and loaded in the DL model.

4. Machine Learning

It is a subset of artificial intelligence that is used for optimization. By using statistical models and computational techniques, it enables models to learn from the given data. However, ML algorithms rely on minimal human intervention to identify features and patterns in the data. To automate the feature extraction process, a subset of ML known as scalable machine learning or deep learning (DL) comes into play. Neural networks are the building blocks of DL algorithm. DL models must have more than three node layers, while a single neural network only needs one. To detect bogus news, this study employs four ML and two DL algorithms. These models are addressed further below.

4.1. Multinomial Naive Bayes

It is a popular supervised learning classification for categorical text data. It is a probabilistic learning method used in NLP. Using the Bayes theorem[31], the method makes a label prediction for a piece of text. It takes a sample and determines the likelihood of each label, then returns the label with the highest likelihood. It considers a feature vector where terms are represented by frequency.

It is based on the following formula:

$$P(A|B) = \frac{P(A)*P(B|A)}{P(B)} \text{..} \quad (i)$$

Where we are calculating the probability of class A when predictor B is already provided.

$P(B)$ =prior probability of B

$P(A)$ =prior probability of class A

$P(B|A)$ =occurrence of predictor B given class A probability

This formula is useful for determining the likelihood of labels within a document.

Pseudo code for binary fake news detection

For Train Split data:

Calculate the class probability (True and Fake)

For Test Split data:

For each given piece of news:

  Initialize probability for each class (True and fake)

  For each word in given piece of news:

Update the probability for each class (True and fake) using Bayes' theorem

The class with the highest probability is predicted as fake or real

4.2. MNB with Pipeline

A pipeline automates different steps to streamline the ML workflow. It enables the concise and efficient formulation of a sequence of data processing stages and a final model. The main goal is to improve accuracy by combining several preprocessing stages and a ML model into a single entity. The pipeline is made up of three main parts:

- The term 'bow' refers to an instance of CountVectorizer, a tool that converts textual input into a document-term matrix.
- The 'tfid' parameter refers to an instance of TfidfTransformer, which is utilized to convert the document-term matrix into the TF-IDF format[1].
- The term 'model' refers to an instance of MultinomialNB, a classifier that embodies the Multinom2 Bayes algorithm.

The pipeline is trained using train split data and is then used to predict the class labels using the test data. The pipeline combines a MNB classifier with text vectorization (CountVectorizer and TfidfTransformer). This assures that text data is processed correctly before being fed into the model, resulting in more reliable outcomes.

4.3. XGBoost with pipeline

Extreme Gradient Boosting (XGBoost) is a gradient boosting algorithm for supervised learning[32]. It builds upon decision trees, ensemble learning, and gradient boosting. The term "gradient boosting" comes from the idea of "boosting" a single weak model by combining it with a number of other weak models to make a strong model. The pipeline utilizes CountVectorizer and TfidfTransformer for text vectorization in addition to an XGBoost classifier.

Pipeline ([

Count Vectorizer= Bag of Words;

TF-IDF Transformer= TF-IDF;

Model= XGB Classifier (

Learning rate=0.1;

Max depth=7;

No of estimators=80;)

])

Max depth, depicts the maximum depth of each tree in XGBoost. No of estimators define the number of boosting rounds.

4.4. Logistic Regression

It is a supervised ML algorithm used for classification into one of the two classes or into one of the many classes[33]. It is a discriminative model because of its ability to discriminate between possible classes. In logistic regression, the logistic or sigmoid function is used to map the linear combination of input features to the range [0, 1].

$x_1, x_2, \dots, x_n$ are the input vectors

Based on features, a straight line, also known as "decision boundary," divides data points into two classes. With the goal of minimizing classification errors, logistic regression determines the decision boundary that most closely matches the data.

The equation for this decision boundary can take various forms:

$y = mx + c$

Where m is the slope, x is the data point, and c is the intercept.

Or $h\theta(x) = \theta_o + \theta_1 x$

Or $h\theta(x) = \beta_o + \beta_1 x$

Or $y = w^t x + b$

Here, w is the weight vector. The goal is to optimize the weight vector to make the cost function $\sum_{i=1}^{n} y_i * w_i^t x_i$ as maximum as possible. This involves iteratively adjusting the values of w_i until we find the best-fit line. The function f is a function that models the relationship between the predicted values.

$Max \sum_{i=1}^{n} f(y_i * w^t x_i)$

The primary aim of binary logistic regression is to develop a model that can effectively make a binary classification decision for a given input observation. Here, the sigmoid function helps to make this decision denoted as

$$\sigma(z) = \frac{1}{1+e^{-z}} \quad \text{(ii)}$$

Where z= $y_i * w^t x_i$

The value close to 0 implies that the news is likely to be real (negative class); a value close to 1 indicates that the given input news is likely to be fake (positive class).

4.5. Convolutional Neural Network (CNN)

CNN maps input data features using convolutional layers. Convolution and pooling are two fundamental operations employed in CNN that serve as mechanisms for extracting features. Different filter sizes are applied to the layers to produce different feature mappings. Based on the feature mapping results, CNN can extract information about the input data. Pairing a convolutional layer with a pooling layer ensures that all filters provide outputs with the same dimensionality. The pooling layer further minimizes the computing effort by minimizing the output dimensions without losing useful data. In the illustration Fig. 5, CNN for text analysis is shown.

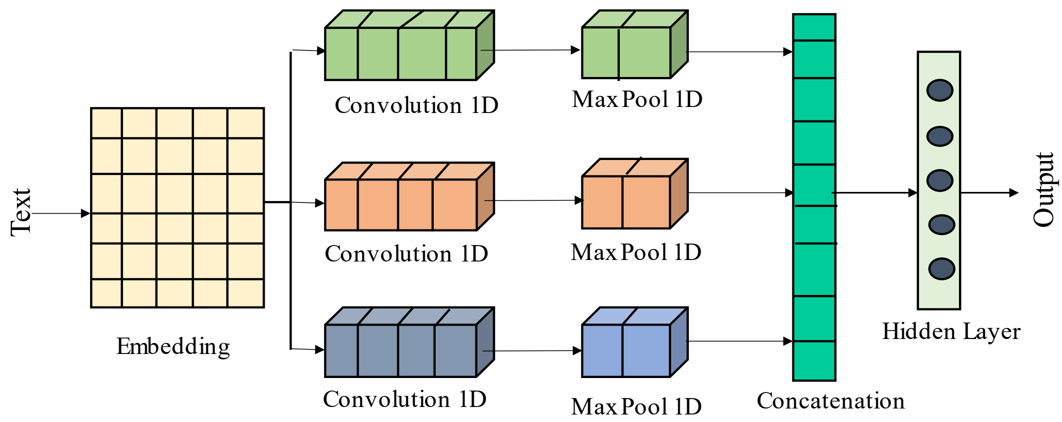

Fig. 5 CNN for text

Word embedding typically employs a convolutional layer, followed by a non-linearity (often ReLU), and finally a pooling operation. These are the fundamental elements of convolutional models; their configurations may vary depending on the task at hand, but they always consist of the same basic parts.

Binary fake news detection using CNN

- $i_1, i_2, i_3, \ldots, i_n$ is input text sequence. where n is the number of tokens
- e_j is the word embedding for each i_j . $e_1, e_2, \ldots, e_n$ is the word embeddings
- For feature extraction apply convolution filter(1D) slides over the input. F is the output feature map

$$F_i = f\left(\sum_{j=1}^{x}(w_j . e_{i+j-1}) + b\right) \quad \text{(iii)}$$

- ➢ Where F_i is the output feature at i_{th} position
- ➢ e_{i+j-1} is the output feature at i_{th} position
- ➢ w_j is the weight vector and b being the bias
- ➢ f is the activation function
- Pooling layer is used to minimize the dimensionality
- Output from the pooling layer is elongated in flattening layer
- Now, it goes through dense layers to make binary predictions (fake or real) on the given input news.

4.6. LSTM

The Long Short-Term Memory (LSTM) network is a RNN variant that is extensively employed in the domain of sequential data prediction tasks[10]. LSTM models have the capability to address the problem of vanishing gradients, a challenge that arises when conventional RNN is trained on extended sequences of data. The input gate is liable for controlling the influx of novel information into the cell, while the forget gate handles what data is discarded. Lastly, the output gate manages the data flow into the output of the LSTM.

Mathematically the functioning of LSTM is shown below:

x_t is the input at time t

i_t is the input gate

f_t is the forget gate

o_t is the output gate

c_t is the current cell or memory state

h_t is the hidden state or output

h_{t-1} is the current cell or memory state

σ depicts the sigmoid activation function

σ_h depicts the tanh activation function

$f_t = \sigma\left(w_f x_t + w_f h_{t-1} + b_f\right)$

$i_t = \sigma(w_i x_t + w_i h_{t-1} + b_i)$

$o_t = \sigma(w_o x_t + w_o h_{t-1} + b_o)$

$c_t = f_t . c_{t-1} + i_t . \sigma_h(w_c x_t + w_c h_{t-1} + b_c)$

$h_t = o_t . \sigma_h(c_t)$

In simple terms, Fig. 6 represents LSTM architecture. It processes sequential data by using input and output gates to control information flow and manage data dependencies. It has a working memory cell for storing short-term information and a long-term memory cell for grasping long-term patterns in data.

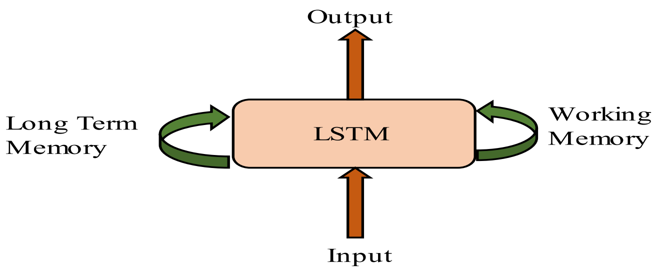

Fig. 6 LSTM Architecture

This process enables the LSTM network to learn sequential patterns in the text and make binary predictions regarding whether the news article is fake or real.

4.7. Evaluation Criteria

To evaluate the research, different metrics Accuracy, Precision, Recall (Sensitivity) and F1 score are used.

Accuracy It calculates the number of accurate detections among all the detections a research model made.

$$\text{Accuracy} = \frac{Number\ of\ right\ detections}{Total\ number\ of\ detections} \text{.......................... (iv)}$$

Precision It measures the accuracy of the positive predictions among all positive predictions made by the model.

$$\text{Precision} = \frac{True_{Positive}}{True_{Positive} + False_{Positive}} \text{........................... (v)}$$

Recall is the measure of how many of the actual positive instances in the corpus are predicted correctly.

$$\text{Recall} = \frac{True_{Positive}}{True_{Positive} + False_{Negative}} \text{............................. (vi)}$$

F1-Score It considers how many of the predicted positive cases are correct (precision) and how many of the actual positive instances found (recall).

$$\text{F1-Score} = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} \text{.................................. (vii)}$$

A high F1 score implies that the model is accurately identifying the target values.

5. Proposed Stacking Approach

Since it is well-known that DL automates the feature extraction process, this research proposes a stacking technique that makes use of both traditional and advanced ML to identify fake news. Algorithm 1 outlines the procedural steps of the suggested method.

Algorithm 1 The steps of the proposed stacking model (CNN+LSTM+LR) for fake news identification

Input: Two labelled fake news datasets;

Output: binary prediction: real or fake news;

Step 1: Load fake news datasets and apply concatenation;

Step 2: Pre-process the merged dataset to make it ready to be input to ML classifiers;

Step 3: Converting the text into numerical vectors;

Step 4: Utilize CNN and LSTM to generate features from text and concatenate the features from both models to create combined feature representations;

Step 5: Input the above and combine feature representations with Logistic Regression;

Step 6: Train the model using training data;

Step 7: Use the train model to make predictions on the testing data;

Step 8: Use classification parameters to measure how well the model works

In order to extract features from the combined corpus, we use two distinct neural network models, namely CNN and LSTM. Then, for binary classification, these features are pooled and fed into the ML technique of Logistic Regression. Using CNN and LSTM layers, a neural network model is constructed to harness the potential of these advanced neural network architectures. Two distinct models are developed and trained to identify false news. First model CNN process the text data to capture the local patterns from it. It consists of an Embedding layer for word representation, a Conv1D layer for capturing local patterns, a MaxPooling1D layer for feature down-sampling. The second model handles text data directly using a LSTM network to extract the sequential dependencies or contextual information from it. The CNN-LSTM model's learnt features are considerably richer than explicitly extracted features, with the ability to reveal latent meanings and temporal relationships. Combining these features and feeding them into a logistic regression model allows to estimate the likelihood that a given instance falls into one of two classes as shown in Fig.7.

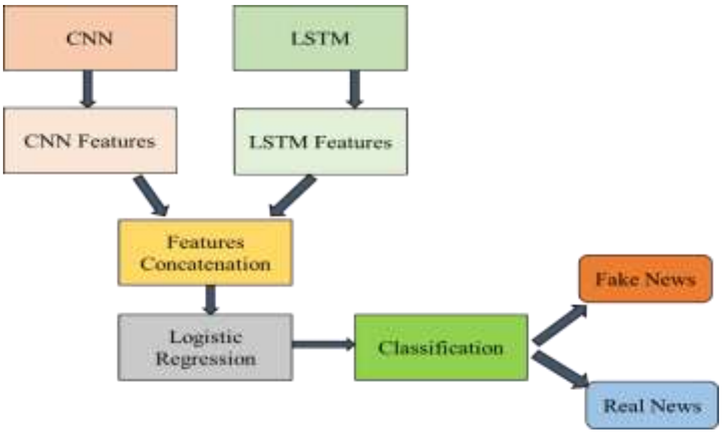

Fig. 7 Pictorial representation of proposed stacking approach

Key parameters used in current work:

Here, a combination of CNN and LSTM models is employed for the classification of news into binary categories. The text data is tokenized with a maximum vocabulary size of 10,000 words and padded to a maximum length of 100. The CNN model used an embedding layer with 10,000 input dimensions, 32 output dimensions, and 100 input sequence lengths. The next layer was a 1D convolutional layer with 32 filters, a kernel size of 5, ReLU activation, and max pooling with a pool size of 4. The CNN model got an LSTM layer with 32 units added to it. It has an embedding layer with the same settings and then an LSTM layer with 32 units. Both models concluded with a dense layer with a single output unit and sigmoid activation. The models were compiled using binary cross-entropy loss and the Adam optimizer. Both the CNN and LSTM models were trained using 32-batch sizes. Afterwards, the features extracted from both models were combined using logistic regression (LR).

Following that, the performance of the proposed model is assessed through a classification report, a standard means of evaluating machine learning models that provides a comprehensive breakdown of their performance across various evaluation criteria (Fig. 8).

	precision	recall	f1-score	support
0	0.93	0.92	0.92	4713
1	0.95	0.96	0.95	7560
accuracy			0.94	12273
macro avg	0.94	0.94	0.94	12273
weighted avg	0.94	0.94	0.94	12273

Fig. 8 Classification report for CNN+LSTM+LR model

The confusion metric is illustrated in Fig. 9 and shows that the proposed strategy yields splendid results, with an accuracy of 94.19%, 95.05% precision, 95.54% recall and 95.29% F1-score.

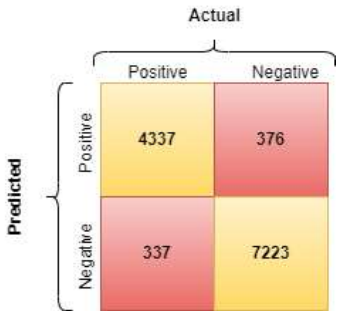

Fig. 9 Confusion matrix for CNN+LSTM+LR

6. Performance Comparison

Several classifiers are used to demonstrate the efficacy of the proposed approach. This section undertakes an evaluation of different ML techniques to ascertain their effectiveness in the identification of false news.

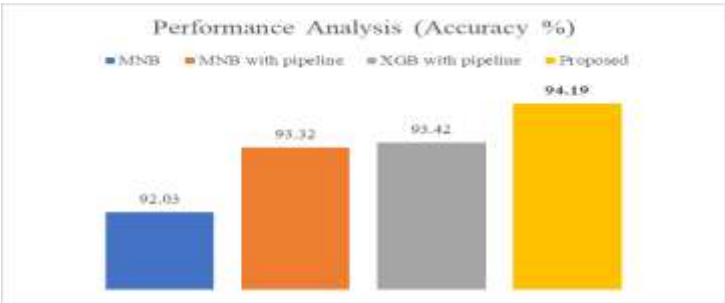

Fig. 10 Performance comparison based on accuracy

Fig. 10 depicts the effectiveness assessment of various classification methods. The multinomial NB algorithm, when applied with TF-IDF, demonstrated a level of accuracy up to 92.03%. However, the use of a pipeline further improved this accuracy to 93.32%. The XGBoost algorithm achieved a pipeline-assisted accuracy rate of 93.42%. The integration of CNN, LSTM, and LR yielded an accuracy of 94.19%. This notable achievement represents substantial progress in fake news identification. Fig. 11 depicts the MNB classifier's confusion matrix.

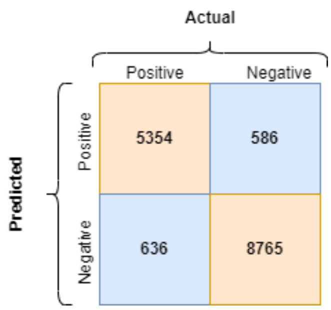

Fig. 11 Confusion matrix for Multinomial NB

Fig. 12 displays the confusion matrix of the MNB classifier with the pipeline of CountVectorizer and TfidfTransformer [25].

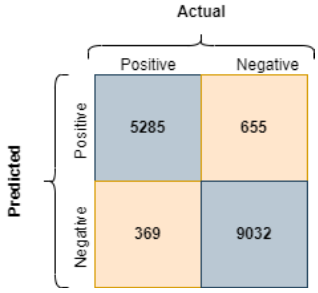

Fig. 12 Confusion matrix for Multinomial NB with pipeline

The high TP and TN values of the multinomial NB classifier exhibit false when the news is false and predict true when the news is actually true. Following Fig. 13 is the classification report of the XGBoost classifier, which we implemented using the pipeline of CountVectorizer and TfidfTransformer.

```
              precision    recall  f1-score   support

           0       0.98      0.85      0.91      4726
           1       0.91      0.99      0.95      7547

    accuracy                           0.93     12273
   macro avg       0.94      0.92      0.93     12273
weighted avg       0.94      0.93      0.93     12273
```

Fig. 13 Classification report for XGB with pipeline

The weighted average F1-score is a statistical measure that calculates the average F1-scores, taking into account the weight assigned to each class based on the number of true cases. The weighted average F1 score for this particular scenario is 0.93. The high precision, recall, and F1-score values for both classes depicted in the classification report indicate that the model performs well with 93.42% accuracy, which is higher than the last implemented approach.

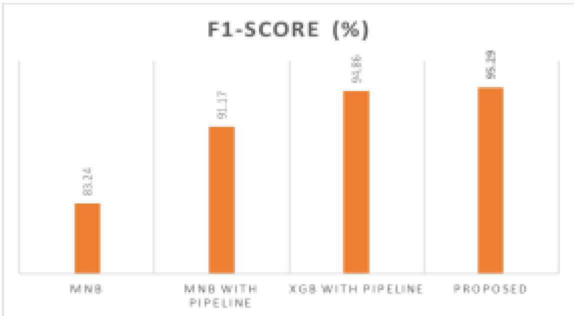

Fig. 14 Model performance comparison based on F1-score

Fig. 14 shows a comparison of the proposed model using the F1-score and shows that it achieved the highest score of 95.29%. This shows that our proposed stacked approach that combines the strengths of different models to achieve balanced performance in identifying fake news works.

7.1 McNemar's Statistical Test

It is a non-parametric statistical test used to analyse paired nominal data. In our study, we use this statistical method to determine the significance of the techniques employed in binary classification of the given news. This test is based on a 2X2 contingency table (Fig. 15) to check the marginal difference of two dichotomous variables.

		Classifier B	
		Correct	Incorrect
Classifier A	Correct	n_{11}	n_{10}
	Incorrect	n_{01}	n_{00}

Fig. 15 Contingency Table.

A dichotomous variable is one that has only two categories. The total number of the samples used in this test is n, where $n = n_{00}+n_{01}+n_{10}+n_{11}$. The McNemar's test ($\chi^2$) can be calculated as follows:

$$\chi 2 = \frac{(|n_{01} - n_{10}| - 1)^2}{n_{01} + n_{10}} \quad \text{(viii)}$$

This formula makes the test more conservative and less likely to overestimate statistical significance. We choose the significance threshold (5%) to determine the rejection or acceptance of the null hypothesis (there is no significant difference between the performance of models). The null hypothesis is rejected if the obtained p-value is less than 0.05, indicating the performance is significant. Table III represents the performance based on statistical test.

Table III Model Performance Evaluation based on Statistical test

Model Comparison	χ2	Significance ($p<0.05$)
MNB vs Proposed	155.51	Yes
XGB vs Proposed	154.37	Yes

The obtained outcome attests to the proposed stacking model's superior efficacy in detecting false news, indicating that the performance improvement is meaningful rather than coincidental.

Various existing methods for detecting fake news are defined in Table IV, which showcases their competitive performance in the evolving landscape of fake news detection.

Table IV Performance comparison with existing techniques

Reference	**Method**	**Dataset**	**Accuracy (%)**
K. Nakamura et al.[16]	BERT	Fakeddit	86.40
S. Raza and C. Ding [17]	BART	Fakeddit	74.80
D. Sharma, S. Garg [23]	Naïve Bayes	IFND	87.50
D. Sharma, S. Garg [23]	LSTM	IFND	92.60
A. Roy et al.[11]	CNN+BiLSTM+MLP	LIAR	44.87
X. Zhi et al. [11]	CNN+LSTM	Self-collected	92.10
Proposed	CNN+LSTM+LR	IFND, FakeNewsIndia	94.19

The above table shows that a variety of factors, such as the quantity and quality of the data, pre-processing techniques, model selection, and parameter selection, influence the performance of model. Data preprocessing ensures that the input data is consistent. The technique selection is another critical factor, as different models have different architectures. Moreover, fine-tuning parameters, such as learning rate, batch size, and activation function, can also significantly affect the performance. As a result, the interplay between these factors determines the model's effectiveness.

7. Conclusion

The emergence of the internet has transformed the way people communicate, and information technology has played a significant role in this revolution. The majority of people now receive their news from the internet, often from unverified sources, which can sometimes be fake news. It can be used to sway public opinion, particularly during elections, to manipulate public opinion about the candidates and the election. It erodes public trust in reliable sources of information, including news organizations and government agencies. Combating fake news is an ongoing challenge. Therefore, detecting and combating fake news is crucial to ensuring that individuals have access to accurate information and to maintaining a healthy and informed society. To contribute in this realm, this study leveraged a combination of CNN, LSTM, and logistic regression for binary textual fake news detection. The fusion of local patterns from CNN and sequential dependencies from LSTM enhances the decision-making Logistic Regression model. With the experiment analysis, it was found that:

- The proposed stacking model attained 94.19% accuracy in fake news detection, which is far better than other hybrid approaches.
- The phenomenal 95.29% F1 score, which is a measure of how well this model can discover bogus news, indicates the remarkable performance achieved with this combination.

By combining the strengths of CNN, LSTM, and LR, the proposed method for detecting bogus news is both efficient and accurate. But still, there is room for improvement. Training data quality and diversity may affect model performance. The loss function, activation function, and optimizer have a considerable impact on determining a task's success rate. However, the focus of this study is on binary classification, and multi-class classification will be investigated in future work. Multi-modal approaches combining various modalities, such as text, picture, audio, and video, to debunk fake news are being investigated as a way forward.

References

[1] V. I. Ilie, C. O. Truica, E. S. Apostol, and A. Paschke, "Context-Aware Misinformation Detection: A Benchmark of Deep Learning Architectures Using Word Embeddings," *IEEE Access*, vol. 9, pp. 162122–162146, 2021, doi: 10.1109/ACCESS.2021.3132502.

[2] W. Ansar and S. Goswami, "Combating the menace: A survey on characterization and detection of fake news from a data science perspective," *Int. J. Inf. Manag. Data Insights*, vol. 1, no. 2, p. 100052, 2021, doi: 10.1016/j.jjimei.2021.100052.

[3] H. A. Khan, "Misinformation is a complex, industry-wide problem, Instagram India's Policy Head says," *February, 07,* 2023. https://www.thehindu.com/sci-tech/technology/misinformation-is-a-complex-industry-wide-problem-instagram-indias-policy-head-says/article66480999.ece

[4] H. Farid, "Seeing Is Not Believing: Doctoring digital photos is easy. Detecting it can be hard.," *IEEE Spectr. 46,* no. august 2009, pp. 44–51, 2009.

[5] J. Y. Khan, M. T. I. Khondaker, S. Afroz, G. Uddin, and A. Iqbal, "A benchmark study of machine learning models for online fake news detection," *Mach. Learn. with Appl.*, vol. 4, no. October 2020, p. 100032, 2021, doi: 10.1016/j.mlwa.2021.100032.

[6] K. M. Fouad, S. F. Sabbeh, and W. Medhat, "Arabic fake news detection using deep learning," *Comput. Mater. Contin.*, vol. 71, no. 2, pp. 3647–3665, 2022, doi: 10.32604/cmc.2022.021449.

[7] R. Oshikawa, J. Qian, and W. Y. Wang, "A survey on natural language processing for fake news detection," *Lr. 2020 - 12th Int. Conf. Lang. Resour. Eval. Conf. Proc.*, no. November 2018, pp. 6086–6093, 2020.

[8] E. Hossain, N. Kaysar, A. Zahid, M. Jalal, U. Joy, and M. M. Rahman, "A Study Towards Bangla Fake News Detection Using Machine Learning and Deep Learning," *Springer*, pp. 79–95, 2022, doi: 10.1007/978-981-16-5157-1_7.

[9] A. D'Ulizia, M. C. Caschera, F. Ferri, and P. Grifoni, "Fake news detection: A survey of evaluation datasets," *PeerJ Comput. Sci.*, vol. 7, pp. 1–34, 2021, doi: 10.7717/PEERJ-CS.518.

[10] X. Zhi *et al.*, "Financial Fake News Detection with Multi fact CNN-LSTM Model," *2021 IEEE 4th Int. Conf. Electron. Technol. ICET 2021*, pp. 1338–1341, 2021, doi: 10.1109/ICET51757.2021.9450924.

[11] A. Roy, K. Basak, A. Ekbal, and P. Bhattacharyya, "A Deep Ensemble Framework for Fake News Detection and Classification," 2018, [Online]. Available: http://arxiv.org/abs/1811.04670

[12] S. Shaar *et al.*, "Overview of the CLEF-2021 CheckThat! Lab Task 2 on detecting previously fact-checked claims in tweets and political debates," *CEUR Workshop Proc.*, vol. 2936, pp. 393–405, 2021.

[13] A. Heidari, N. Jafari Navimipour, M. Unal, and S. Toumaj, "The COVID-19 epidemic analysis and diagnosis using deep learning: A systematic literature review and future directions," *Comput. Biol. Med.*, vol. 141, no. November 2021, p. 105141, 2022, doi: 10.1016/j.compbiomed.2021.105141.

[14] J. C. S. Reis, A. Correia, F. Murai, A. Veloso, F. Benevenuto, and E. Cambria, "Supervised Learning for Fake News Detection," *IEEE Intell. Syst.*, vol. 34, no. 2, pp. 76–81, 2019, doi: 10.1109/MIS.2019.2899143.

[15] L. Alzubaidi *et al.*, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *J. Big Data*, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-021-00444-8.

[16] K. Nakamura, S. Levy, and W. Y. Wang, "r/Fakeddit: A new multimodal benchmark dataset for fine-grained fake news detection," *Lr. 2020 - 12th Int. Conf. Lang. Resour. Eval. Conf. Proc.*, pp. 6149–6157, 2020, Accessed: May 20, 2022. [Online]. Available: https://www.journalism.org/2019/06/05/many-americans-

[17] S. Raza and C. Ding, "Fake news detection based on news content and social contexts: a transformer-based approach," *Int. J. Data Sci. Anal.*, vol. 13, no. 4, pp. 335–362, 2022, doi: 10.1007/s41060-021-00302-z.

[18] H. M. Jabir, M. A. Naser, and S. O. Al-Mamory, "Rumor Detection on Twitter Using Features Extraction Method," *Proc. 2020 1st Inf. Technol. to Enhanc. E-Learning other Appl. Conf. IT-ELA 2020*, pp. 115–120, 2020, doi: 10.1109/IT-ELA50150.2020.9253027.

[19] E. Tacchini, G. Ballarin, M. L. Della Vedova, S. Moret, and L. de Alfaro, "Some like it Hoax: Automated fake news detection in social networks," *CEUR Workshop Proc.*, vol. 1960, pp. 1–12, 2017.

[20] N. Aslam, I. Ullah Khan, F. S. Alotaibi, L. A. Aldaej, and A. K. Aldubaikil, "Fake Detect: A Deep Learning Ensemble Model for Fake News Detection," *Complexity*, vol. 2021, 2021, doi: 10.1155/2021/5557784.

[21] H. Saleh, A. Alharbi, and S. H. Alsamhi, "OPCNN-FAKE: Optimized Convolutional Neural Network for

Fake News Detection," *IEEE Access*, vol. 9, pp. 129471–129489, 2021, doi: 10.1109/ACCESS.2021.3112806.

[22] K. Verma *et al.*, "Latest tools for data mining and machine learning," *Int. J. Innov. Technol. Explor. Eng.*, vol. 8, no. 9 Special Issue, pp. 18–23, 2019, doi: 10.35940/ijitee.I1003.0789S19.

[23] D. K. Sharma and S. Garg, "IFND: a benchmark dataset for fake news detection," *Complex Intell. Syst.*, 2021, doi: 10.1007/s40747-021-00552-1.

[24] A. Dhawan, M. Bhalla, D. Arora, R. Kaushal, and P. Kumaraguru, "FakeNewsIndia: A benchmark dataset of fake news incidents in India, collection methodology and impact assessment in social media," *Comput. Commun.*, vol. 185, no. January, pp. 130–141, 2022, doi: 10.1016/j.comcom.2022.01.003.

[25] P. Mahajan and J. Kaushal, "Epidemic Trend of COVID-19 Transmission in India During Lockdown-1 Phase," *J. Community Health*, vol. 45, no. 6, pp. 1291–1300, 2020, doi: 10.1007/s10900-020-00863-3.

[26] S. Banaji and R. Bhat, "WhatsApp Vigilantes: An exploration of citizen reception and circulation of WhatsApp misinformation linked to mob violence in India," *LSE Media Commun.*, vol. 2, pp. 1–14, 2020, [Online]. Available: https://www.lse.ac.uk/media-and-communications/assets/documents/research/projects/WhatsApp-Misinformation-Report.pdf

[27] S. Shekhar, H. Garg, R. Agrawal, S. Shivani, and B. Sharma, "Hatred and trolling detection transliteration framework using hierarchical LSTM in code-mixed social media text," *Complex Intell. Syst.*, vol. 9, no. 3, pp. 2813–2826, 2023, doi: 10.1007/s40747-021-00487-7.

[28] I. K. Sastrawan, I. P. A. Bayupati, and D. M. S. Arsa, "Detection of fake news using deep learning CNN–RNN based methods," *ICT Express*, vol. 8, no. 3, pp. 396–408, 2022, doi: 10.1016/j.icte.2021.10.003.

[29] M. F. Mridha, A. J. Keya, M. A. Hamid, M. M. Monowar, and M. S. Rahman, "A Comprehensive Review on Fake News Detection with Deep Learning," *IEEE Access*, vol. 9, pp. 156151–156170, 2021, doi: 10.1109/ACCESS.2021.3129329.

[30] T. Jiang, J. P. Li, A. U. Haq, A. Saboor, and A. Ali, "A Novel Stacking Approach for Accurate Detection of Fake News," *IEEE Access*, vol. 9, pp. 22626–22639, 2021, doi: 10.1109/ACCESS.2021.3056079.

[31] M. Fayaz, A. Khan, M. Bilal, and S. U. Khan, "Machine learning for fake news classification with optimal feature selection," *Soft Comput.*, vol. 26, no. 16, pp. 7763–7771, 2022, doi: 10.1007/s00500-022-06773-x.

[32] J. A. Nasir, O. S. Khan, and I. Varlamis, "Fake news detection: A hybrid CNN-RNN based deep learning approach," *Int. J. Inf. Manag. Data Insights*, vol. 1, no. 1, p. 100007, 2021, doi: 10.1016/j.jjimei.2020.100007.

[33] S. U. Hassan, J. Ahamed, and K. Ahmad, "Analytics of machine learning-based algorithms for text classification," *Sustain. Oper. Comput.*, vol. 3, no. April, pp. 238–248, 2022, doi: 10.1016/j.susoc.2022.03.001.