# A Novel Voting Ensemble Stacking Approach for Fake News Detection

E. Prabhakar
Assistant Professor
Department of Computer Science and Engineering
Nandha College of Technology
Erode, India

Venkatesh.J
UG - Final year
Department of Computer Science and Engineering
Nandha College of Technology
Erode, India

Sathyan.T
UG - Final year
Department of Computer Science and Engineering
Nandha college of Technology
Erode, India

Karthik.S
UG- Final year
Department of Computer Science and Engineering
Nandha college of Technology
Erode, India.

Vijayaragavan.D
UG-Final year
Department of Computer Science and Engineering
Nandha college of Technology
Erode, India.

*Abstract*— The detection of fake news is a critical task in today's digital age and misinformation spreads rapidly. This paper presents a novel ensemble stacking approach for the detection of fake news. LIAR dataset is used for this approach. The dataset is subjected to data cleaning and data pre-processing to enhance accuracy and efficiency. Natural Language Processing concepts such as Bag-of-Words (BoW) and Term Frequency – Inverse Document Frequency (TF-IDF) were employed to convert text into numerical data. Machine Learning models such as Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbour Embedding (t-SNE) were utilized for exploring and visualizing complex high-dimensional data. The pre-processed data is fed to six Machine Learning classifiers, one Deep Learning classifier and two combinations of Machine and Deep learning classifiers. Machine learning algorithms such as Logistic Regression, Naive Bayes, Decision Tree, Random Forest, Support Vector Machine, and Adaptive Boosting were used. Deep Learning model called Multi-Layer Perceptron was used. Combination of MLP with PCA and MLP with t-SNE were used. The individual models were evaluated by precision, recall, f1-score, support metrics. The trained models were ensemble stacked with Novel Voting Ensemble as a final classifier to predict whether the news is fake or real. Our approach combines various approaches and contributes to combating the spread of misinformation and ensuring the integrity of news sources in the digital era.

*Keywords*—Fake news detection, Machine learning algorithms, Data preprocessing, Natural language processing (NLP), LIAR dataset, Ensemble stacking, Feature extraction, Dimensionality reduction

## I. INTRODUCTION

In modern digitalized world, the proliferation of fake news presents a significant challenge to the credibility of information sources and also pose a significant threat to sovereignty and social welfare of the nation. Detecting fake news accurately and efficiently is crucial for maintaining public order and ensuring the dissemination of accurate information. This paper proposes a novel ensemble stacking approach to tackle the problem of fake news detection.

To ensure effective detection, concepts such as ML, DL and NLP were used. The methodology begins with meticulous data preprocessing to enhance the quality of input data. In data cleaning, rows with inadequate information and unwanted columns were removed. The data is pre-processed with binary classification, conversion into lowercase, removal of punctuations, tokenization, lemmatization, removal of stop words and finally vectorized with BoW and t-SNE. The complexity in high dimensional data is simplified by PCA algorithm. Then the data is fed to t-SNE for data exploration and visualizing high dimensional data. The classifiers such as Logistic Regression, Naive Bayes, Decision Tree, Random Forest, Neural Networks, Support Vector Machine, Adaptive boosting, MLP, MLP with PCA, MLP with t-SNE were trained. Through ensemble stacking, our approach combines the strengths of individual classifiers to achieve superior performance in distinguishing between genuine and fake news articles. By harnessing the collective intelligence of multiple models, the accuracy and efficiency in fake news detection, was so high, thereby contributing to the integrity of information dissemination in the digital realm.

## II. LITERATURE SURVEY

Various scholars have recommended numerous machine learning and deep learning techniques for detecting fake news. This research study outlines several baseline methods for fake news detection. The primary aim of the literature is to pinpoint the shortcomings in these baseline approaches and propose effective

solutions. Some researchers have extensively assessed various machine learning models across diverse datasets to determine the optimal individual model. Singh et al. [1] investigated the serious problem of fake news on social media and emphasized that early detection is significant in major areas like politics and education. Dnyandeo Waghmare et al. [2] conducted the experimental analysis on the LIAR dataset to evaluate the classification accuracy of fake news detection.

Mondal et al. [3] proposed the importance of text preprocessing techniques which enables to comprehend the characteristics of various entities of a given dataset like column(s), row(s), etc. in building accurate classifiers for fake news detection. Narkhede et al. [4] describes the need of libraries in efficient data pre-processing. Pendyala et al. [5] used spectral methods to analyse how fake information propagates and being unresolved. Their analysis indicates that how fake news is closely interconnected with authentic information and the machine learning algorithms are not as proficient in distinguishing between the two. Gao et al. [6] utilization of the Naive Bayes algorithm, which employs the principles of Bayes' Theorem and conditional independence assumptions to calculate and compare probabilities for classification purposes, learned that the use of Naïve bayes model have many advantages of reducing complexity, consistent classification, etc. So, implementing Naïve bayes algorithm involves higher accuracy of categorization real and fake news.

Warjri et al. [7] employed Logistic Regression, Decision Tree classifier for solving fake social media news in Khasi language and results shows that Decision Tree performed top notch when using language datasets. Llugsi et al. [8] used Hyperparameter Optimization Strategy to tweak model performance for optimal results and enabled us to learn, choosing the accurate hyperparameter values is a crucial process for the success of Machine Learning algorithms. Agarwal et al. [9] employed machine learning models on LIAR dataset, founded that TF-IDF outperformed others. Qasem et al. [10] analysed the N-grams, which represents the number of elements in the sequence, utilized for contextual understanding. Santhiya et al. [11] discussed about three ML approaches like SVM, RF, NB Classifiers to determine the genuineness of a employment advertisement. By use of voting method, finally it gives good accuracy.

Jung et al. [12] discussed Projection Ensemble approach and its advantages of robust structure identification, enhanced visualization, reduced chance of erroneous findings and potential for consensus building. Iqbal et al. [13] studied and presented a novel approach that identifies non-genuine reviews. He used amazon review dataset is used. when Comparing classifiers, Random Forest performed well due to the ability to amalgamate the output of multiple decision trees into a unified outcome. Roshinta et al. [14] used the metrics like accuracy, recall, precision, and F1–score to evaluate the models. Krishna et al. [15] developed a system employing the DT, juxtaposed with SVM which determines the optimal hyperplane to partition the space into two sub-phases- one for vectors belonging to specified category and another for those that do not. Kulkarni et al. [16] used Supervised Machine Learning techniques to detect fake news, particularly SVM and ensemble learning methods like Adaboost, etc. Comparison shows that Adaboost gives higher accuracy rate when used with SVM techniques.

Implementing a robust system for fake news detection involves clear-cut understanding of the Machine and Deep Learning [17] [18] [19] [20] [21] [22] as envisaged in these research studies.

## III. EXISTING APPROACH AND THEIR DRAWBACKS

The existing approaches for fake news detection encompass a variety of techniques, including language pattern analysis, source credibility assessment, user behaviour examination, social network analysis, machine learning models, semantic analysis, and fact-checking integration. However, these methods suffer from several drawbacks such as biases in training data, overemphasis on sources, limited applicability to specific datasets, privacy concerns related to user behaviour analysis, absence of multiple technique integration, and challenges regarding timeliness and standardization. These limitations highlight the need for novel approaches to address the shortcomings of current fake news detection systems.
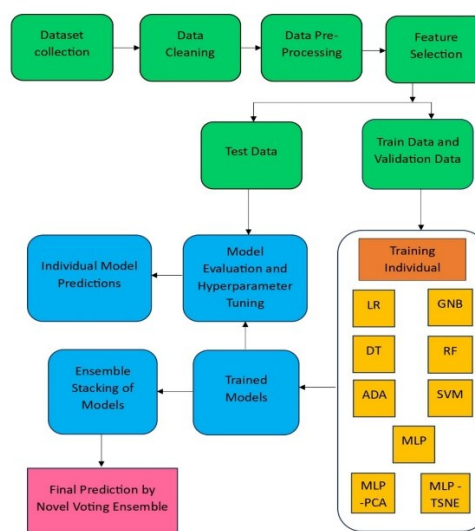
## IV. PROPOSED METHODOLOGY

### A. Workflow



Fig 1. Workflow

### B. Dataset

The LIAR dataset, an necessary component of our fake news detection project, comprises approximately 12.8K concise news statements gathered over a decade from POLITIFACT.COM, a reputable fact-checking platform. Each statement underwent meticulous labelling by human annotators to ensure accuracy. Each statement is renowned for its in-depth scrutiny of political claims, providing reliable data. It categorizes statements into six types—True, Mostly True, Half True, Barely True, False, and Pants on Fire—and includes metadata such as subject, speaker, job title, state, party affiliation, and context. With its diverse content, this dataset facilitates thorough training and assessment of fake news detection models.
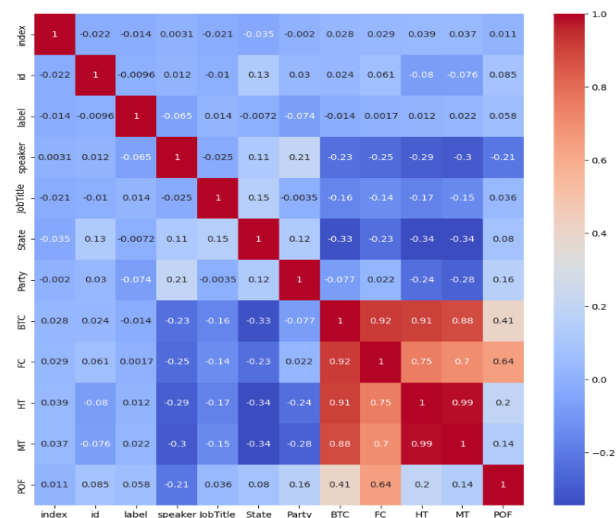
Fig 2. Dataset Representation

C. Data Preprocessing

The data preprocessing phase involves several crucial steps to prepare the raw data for analysis:

*1)* Data Cleaning: Raw data often has inconsistencies, missing values, or outliers that can adversely affect the performance of machine learning models. The data cleaning techniques such as handling missing values, dropping non-required columns, and removing rows with missing data were applied to ensure the integrity and quality of the dataset.

2) Text Preprocessing: Since the data primarily consists of textual information, extensive text preprocessing techniques were employed to transform the raw text into a format suitable for analysis:

- Converting Data into Binary Classification: The dataset was transformed into a binary classification task, distinguishing between genuine and fake news.
- Merging Statement and Subject Columns: The statement and subject columns were merged into a single text column to capture the complete context of each entry.
- Converting Data into Lowercase: All text data was converted into lowercase to standardize the text format and reduce redundancy.
- Removing Punctuation Except Commas: Punctuation marks were removed from the text data, except for commas, to ease further processing.
- Tokenization: The text data was tokenized into individual words to break down the text into meaningful units.
- Lemmatization: Lemmatization was applied to reduce words to their base or root form, ensuring consistency in word representation.
- Stop words Removal: Common stop words were removed from the text data to drop noise and focus on significant terms.
- Joining Text Column into String: The processed text columns were concatenated into a single string for vectorization

3) Feature Vectorization: Following text preprocessing, two popular NLP techniques such as Bag-of-Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF), were employed for feature vectorization:

- Bag-of-Words (BoW): BoW representation was used to convert the pre-processed text data into numerical features, representing each document as a vector of word counts.
- Term Frequency-Inverse Document Frequency (TF-IDF): TF-IDF representation was utilized to calculate the importance of each word in a document relative to the entire corpus, capturing the relevance of words across documents.
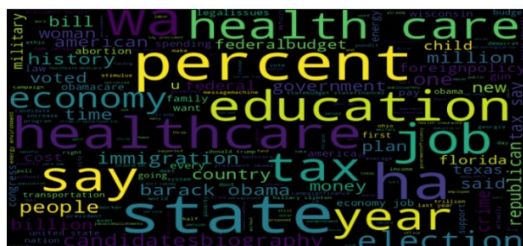


Fig 3. Feature Vectorization

4) Dimensionality Reduction: High-dimensional data can pose challenges for Machine Learning algorithms. Dimensionality reduction Unsupervised Machine Learning algorithms such as Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbour Embedding (t-SNE) were applied to reduce the dimensionality of the feature space while preserving the needed structure and relationships within the data.
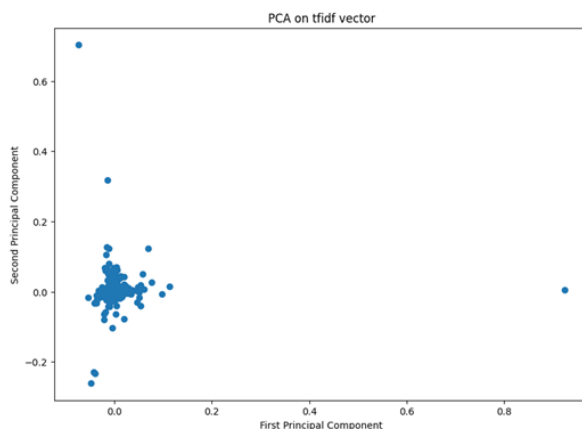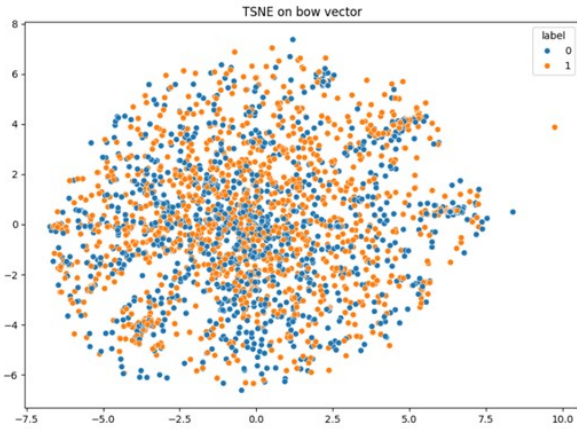


Fig 4. PCA on TF-IDF vector

Fig 5. TSNE on Bow Vector

### D. Classification Models

Six Machine Learning classifiers and One Deep Learning classifiers were employed for the classification task of detecting fake news. These classifiers were trained on pre-processed textual data and aim to distinguish between genuine and fake news articles based on various features extracted from the text. The Seven classification models employed, as follows.

*1)* Logistic Regression*:* Logistic Regression is a linear classification algorithm that predicts the probability of a binary outcome. It is widely used for binary classification tasks and works well with large datasets. The logistic regression formula is:

$$P(Y = 1) = \frac{1}{1+e^{-(b_0+b_1X_1+b_2X_2+\cdots+b_nX_n)}} \qquad (1)$$

Here:

- $P(Y = 1)$ is the probability of the dependent variable being 1.
- $e$ is the base of the natural logarithm. $b_0$ is the intercept.
- $b_1, b_2, \dots, b_n$ are the coefficients for the independent variables $X_1, X_2, \dots, X_n$, respectively.

*2)* Gaussian Naive Bayes: Naive Bayes, rooted in Bayes' theorem, assumes strong feature independence, making it suitable for text classification, especially with high-dimensional data. The Gaussian Naive Bayes assumes features follow a normal distribution, with its probability density function (PDF) used in this formula:

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} e^{-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}} \qquad (2)$$

Here:

- $P(x_i \mid y)$ is the probability of feature $x_i$ given class $y$.
- $\pi$ is the mathematical constant Pi.
- $\sigma_y^2$ is the variance of feature $x_i$ for class $y$.
- $\mu_y$ is the mean of feature $x_i$ for class $y$.
- $e$ is the base of the natural logarithm.

In practice, for a given data point with features $x_1, x_2, \dots, x_n$, the class prediction is determined by:

$$P(y \mid x_1, x_2, \dots, x_n) \propto P(y) \prod_{i=1}^{n} P(x_i \mid y) \qquad (3)$$

Where $P(y)$ is the prior probability of class $y$.

*3)* Decision Tree*:* The Decision Tree, a non-linear classification method, divides data into subsets based on key features. It's interpretable and handles both numerical and categorical data by using feature conditions to predict outcomes. The formula for predicting the target variable in a decision tree is a series of if-else conditions.

Given a decision tree with nodes, branches, and leaves:

1. Each node represents a feature/attribute and a condition.
2. Each branch represents the outcome of the condition (true or false).
3. Each leaf node represents the predicted outcome or class.

The decision-making process follows a path from the root to a specific leaf based on the feature conditions. For example, for a binary decision tree: If $X_1 \leq$ threshold $_1$ and $X_2 >$ threshold $_2$, then predict Class A, else predict Class B. Here:

- $X_1, X_2$ are features.
- threshold $_1$, threshold $_2$ are threshold values for the respective features.

This process continues recursively until a leaf node is reached, providing the final prediction.

*4)* Random forest*:* Random Forest is an ensemble learning method that combines predictions from multiple decision trees, reducing overfitting and enhancing model performance. Each tree in the forest casts a vote, with the final prediction determined by majority vote or averaging. For a classification task, the formula for the predicted class probability $P($ Class $\mid \mathbf{X})$ in a Random Forest is often represented as follows:

$$P(\text{ Class } \mid \mathbf{X}) = \frac{1}{N}\sum_{i=1}^{N} P_i(\text{ Class } \mid \mathbf{X}) \qquad (4)$$

Here:

- $N$ is the number of trees in the Random Forest.
- $P_i($ Class $\mid \mathbf{X})$ is the predicted class probability from the $i$-th decision tree.

For a regression task, the formula for the predicted output in a Random Forest is:

$$\hat{Y}(\mathbf{X}) = \frac{1}{N}\sum_{i=1}^{N} \hat{Y}_i(\mathbf{X}) \qquad (5)$$

Here:

- $\hat{Y}_i(\mathbf{X})$ is the predicted output from the $i$ th decision tree.

In both cases, the Random Forest combines the predictions from multiple trees to provide a more robust and accurate overall prediction. The individual decision trees are trained on different subsets of the data using bootstrapped samples and random feature subsets to promote diversity within the ensemble.

*5) ADABoost Classifier:* ADABoost is an ensemble method that combines weak classifiers to build a strong one, emphasizing misclassified instances for improved performance. The classifier assigns weights to data points and combines weak classifiers to make predictions, determining the final outcome.

For a binary classification problem:

1. Initialize the weights $w_i$ for each data point to be $1/N$, where $N$ is the number of data points.
2. For $t = 1$ to $T$, where $T$ is the number of weak classifiers:
   a. Train a weak classifier $h_t$ using the weighted data.
   b. Calculate the error $\epsilon_t$, which is the weighted sum of misclassified points.
   c. Calculate the classifier weight $\alpha_t = \frac{1}{2}\ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$.
   d. Update the weights $w_i$ based on whether the data point $x_i$ was classified correctly or not.
3. The final prediction is the weighted sum of the weak classifiers' predictions:

Final Prediction $(x) = \text{sign}\left(\sum_{t=1}^{T}\alpha_t h_t(x)\right)$

Here:
- $h_t(x)$ is the prediction of the $t$-th weak classifier.
- $\alpha_t$ is the weight assigned to the $t$-th weak classifier.
- The sum is taken over all weak classifiers.
- sign $(\cdot)$ function returns the sign of the argument.

*6) Support Vector Machine (SVM):* SVM constructs hyperplanes to classify data in high-dimensional spaces, suitable for linear and non-linear tasks. The decision function seeks the best hyperplane to separate classes in the feature space. For a binary classification problem, the formula of the decision function is:

$$f(x) = \text{sign}\left(\sum_{i=1}^{N}\alpha_i y_i K(x, x_i) + b\right) \quad (6)$$

Here:

- $f(x)$ is the decision function for predicting the class of a new data point $x$.
- $N$ is the number of support vectors.
- $\alpha_i$ are the Lagrange multipliers (coefficients) associated with each support vector.
- $y_i$ is the class label of the $i$-th support vector.
- $K(x, x_i)$ is the kernel function that computes the similarity between $x$ and the $i$-th support vector $x_i$.
- $b$ is the bias term.

The sign function returns +1 for positive values and -1 for negative values, effectively classifying the data point based on the decision function result. The SVM aims to find the optimal values for $\alpha_i$ and $b$ during the training process. The choice of the kernel function influences how the SVM captures the relationship between data points in the feature space.

*7) MLP (Multi-Layer Perceptron):* The Multi-Layer Perceptron (MLP) constitutes a variant of artificial neural networks featuring several layers, including an input layer, multiple hidden layers, and an output layer. The formula for the output of an MLP can be expressed as,

$$a_j^{(l)} = \sigma\left(\sum_{i=1}^{n^{(l-1)}} w_{ij}^{(l)} a_i^{(l-1)} + b_j^{(l)}\right) \quad (7)$$

Here:

- $a_j^{(l)}$ is the activation of the $j$-th neuron in layer $l$.
- $\sigma(\cdot)$ is the activation function applied element-wise.
- $n^{(l-1)}$ is the number of neurons in the previous layer.
- $w_{ij}^{(l)}$ is the weight connecting the $i$-th neuron in layer $l-1$ to the $j$-th neuron in layer $l$.
- $a_i^{(l-1)}$ is the activation of the $i$-th neuron in layer $l-1$.
- $b_j^{(l)}$ is the bias term for the $j$-th neuron in layer $l$.

For the output layer $(L)$, the final prediction $\hat{y}_k$ for class $k$ is obtained by applying the softmax activation:

$$\hat{y}_k = \frac{e^{a_k^{(L)}}}{\sum_{j=1}^{C} e^{a_j^{(L)}}} \quad (8)$$

Here:

- $C$ is the number of classes.

*8) MLP with PCA (Principal Component Analysis):* The combination of Multi-Layer Perceptron (MLP) and Principal Component Analysis (PCA) involves using PCA to preprocess the data before feeding it into the MLP. The general formula for this process can be outlined in a series of steps:

1. Standardize the data:

$$X_{\text{std}} = \frac{(X-\mu)}{\sigma} \quad (9)$$

Here:

- $X_{\text{std}}$ is the standardized data.
- $X$ is the original data.
- $\mu$ is the mean of each feature.
- $\sigma$ is the standard deviation of each feature.

4. Compute the PCA transformation: Use PCA to transform the standardized data into a reduced dimensional space:

$$X_{\text{PCA}} = X_{\text{std}} W \quad (10)$$

Here:
- $X_{\text{PCA}}$ is the data transformed into the reduced-dimensional space.
- $W$ is the projection matrix.
3. Feed the PCA-transformed data into the MLP: Use the transformed data $X_{\text{PCA}}$ as the input to the MLP. For the MLP, the forward pass formula remains the same as mentioned earlier:

$$a_j^{(l)} = \sigma\left(\sum_{i=1}^{n^{(l-1)}} w_{ij}^{(l)} a_i^{(l-1)} + b_j^{(l)}\right) \tag{11}$$

And for the output layer, if it's a classification task, the SoftMax activation can be applied. The idea is that PCA is used to reduce the dimensionality of the data before training the MLP. This can be beneficial in cases where the original feature space is large, and PCA helps in capturing the most important aspects of the data before passing it through the neural network.

9) MLP (t-SNE): The integration of Multi-Layer Perceptron (MLP) with t-Distributed Stochastic Neighbor Embedding (t-SNE) involves using t-SNE to preprocess the data before feeding it into the MLP. The overall process can be outlined in several steps:

1. Standardize the data:

$$X_{\text{std}} = \frac{(X - \mu)}{\sigma} \tag{12}$$

   Here:
   - $X_{\text{std}}$ is the standardized data.
   - $X$ is the original data.
   - $\mu$ is the mean of each feature.
   - $\sigma$ is the standard deviation of each feature.

2. Compute the t-SNE transformation: Use t-SNE to transform the standardized data into a lowdimensional space:

$$Y_{\text{t-SNE}} = \text{t-SNE}(X_{\text{std}}) \tag{13}$$

3. Feed the t-SNE-transformed data into the MLP: Use the transformed data $Y_{\text{t-SNE}}$ as the input to the MLP. For the MLP, the forward pass formula remains the same as mentioned earlier:

$$a_j^{(l)} = \sigma\left(\sum_{i=1}^{n^{(l-1)}} w_{ij}^{(l)} a_i^{(l-1)} + b_j^{(l)}\right) \tag{14}$$

And for the output layer, if it's a classification task, the SoftMax activation can be applied. The idea is that t-SNE is used to reduce the dimensionality of the data before training the MLP. t-SNE focuses on preserving local similarities, which can be beneficial in capturing the intricate structures of the data before passing it through the neural network.

E. Ensemble Approach

In our proposed fake news detection system, we employ an ensemble stacking approach to enhance the accuracy and reliability of the classification process. Ensemble learning techniques have demonstrated effectiveness in various domains by leveraging the collective intelligence of multiple base models.

1) Base Model Selection: We begin by selecting a diverse set of base classification models, including logistic regression, naive Bayes, decision tree, random forest, neural networks, support vector machine, and ADABoost classifier. Each model offers unique strengths in capturing different aspects of the underlying data, contributing to the diversity of perspectives in the ensemble.

2) Model Training and Prediction: The selected base models are trained on the preprocessed dataset to learn the underlying patterns indicative of fake or genuine news articles. Once trained, each base model independently predicts the class labels for new instances based on its learned parameters and features.

3) Meta-Model Construction: The predictions generated by the base models serve as inputs to a meta-model, which combines these predictions to make the final classification decision. We employ a stacking architecture, where the outputs of the base models are used as features for training the meta-model. This hierarchical structure allows the meta-model to learn the optimal way of aggregating predictions from diverse base models.

4) Evaluation and Performance Metrics: We evaluate the performance of the ensemble approach using standard metrics such as accuracy, precision, recall, and F1-score. By comparing the performance of the ensemble model with individual base models, it assess the effectiveness of ensemble learning in improving fake news detection accuracy.

```
Classification Report on Validation Set:
              precision    recall  f1-score   support

           0       0.78      0.81      0.80       161
           1       0.86      0.83      0.85       220

    accuracy                           0.82       381
   macro avg       0.82      0.82      0.82       381
weighted avg       0.83      0.82      0.82       381
```

FIGURE 6. Evaluation Metrices

5) Visualization and Interpretation: To provide insights into the decision-making process of the ensemble model, we visualize the contribution of each base model to the final prediction using techniques such as feature importance plots and decision boundaries. These visualizations help in understanding the strengths and weaknesses of individual models and their collective impact on the overall performance.

6) Integration of Results: Finally, we integrate the predictions generated by the ensemble model into our fake news detection system, allowing users to make informed decisions about the credibility of news articles. The ensemble approach enhances the system's reliability and robustness by combining complementary insights from diverse models, thereby mitigating the spread of misinformation.
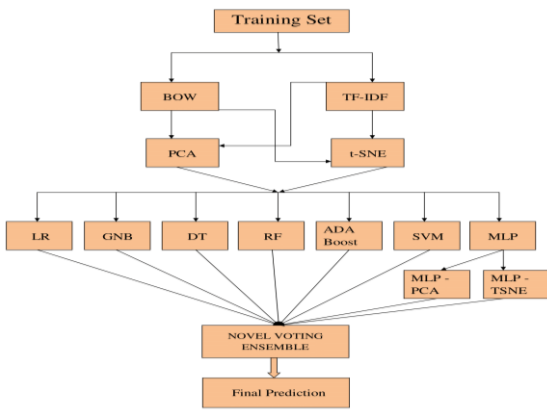
7) Representation of Ensemble approach:



Fig 7. Ensemble Approach

f.  Advantages

*1*) Enhanced Detection Accuracy*:* The ensemble stacking approach amalgamates the strengths of multiple classification models, mitigating individual model biases and errors. By leveraging diverse perspectives on the dataset, our methodology achieves superior accuracy,  compared to standalone models.

*2*) Robustness to Dataset Variability: The utilization of various classification algorithms ensures robustness to dataset variability and inherent noise. The ensemble framework adapts to diverse linguistic patterns and content characteristics, enhancing generalization performance across different news sources and topics.

*3*) Improved Generalization*:* Ensemble learning reduces overfitting by aggregating predictions from multiple base models, resulting in enhanced generalization to unseen data. The meta-model effectively combines diverse model outputs, minimizing the risk of erroneous classifications on new instances.

*4*) Flexibility and Scalability*:* Our methodology supports the integration of additional base models or feature engineering techniques, facilitating adaptation to evolving data characteristics and detection requirements. This flexibility enhances scalability, enabling seamless integration with diverse news sources and languages.

*5*) Interpretability and Transparency*:* Visualization techniques, such as feature importance plots and decision boundaries, provide insights into the decision-making process of the ensemble approach. This transparency enhances interpretability, fostering user trust and understanding of the fake news detection system.

*6*) Comprehensive Evaluation*:* Performance evaluation metrics, including accuracy, precision, recall, and F1-score, enable comprehensive assessment of the proposed methodology. Comparative analysis with individual models validates the efficacy of ensemble learning in improving detection performance.

*7*) *Practical Utility:* The integration of ensemble predictions into a practical fake news detection system empowers users to

evaluate news credibility effectively. By leveraging diverse model insights, our methodology aids in combating misinformation dissemination and promoting informed decision-making.

## IV. RESULT OF ENSEMBLE APPROACH

The performance of the ensemble stacking approach was evaluated using various metrics to assess its efficacy in fake news detection. The experimental results demonstrate the superiority of the proposed methodology compared to individual classification models.
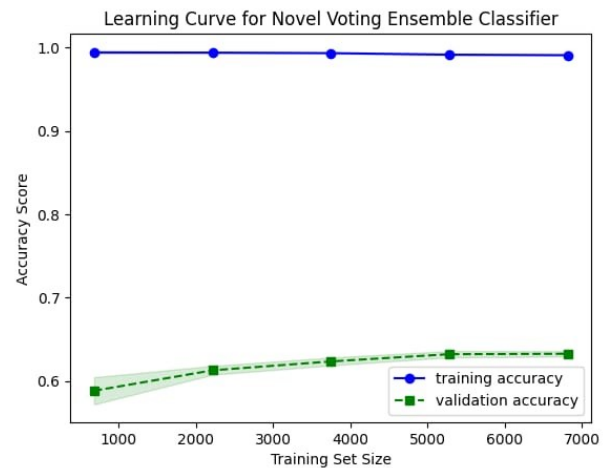


Fig 8. Result of Ensemble approach

TABLE 1. ACCURACY OF MODELS

| MODELS | ACCURACY (IN PERCENTAGE) |
|---|---|
| Logistic Regression Classifier | 82 |
| Naïve Bayes Classifier | 68 |
| Decision Tree Classifier | 82 |
| Random Forest Classifier | 83 |
| Adaboost Classifier | 64 |
| Support Vector Machine Classifier | 82 |
| Multi-Layer Perceptron Classifier | 85 |
| Mlp With Principal Component Analysis Classifier | 61 |
| Mlp With T-Distributed Stochastic Neighbor Embedding Classifier | 68 |
| Novel Voting Ensemble Classifier | 88 |

## V. CONCLUSION

In this research, a novel voting ensemble stacking approach was developed for the accurate detection of fake news, leveraging a combination of machine learning and natural language processing techniques. The proposed methodology demonstrated promising results, outperforming individual classification models and baseline approaches in terms of accuracy and reliability. By integrating multiple classifiers within a layered structure, the

ensemble model effectively captured diverse patterns and features indicative of fake news, thereby enhancing detection efficacy. Furthermore, extensive experimentation and evaluation validated the robustness and generalization capability of the ensemble approach across different datasets and scenarios. Overall, this research contributes to the advancement of fake news detection systems, fostering trust and credibility in online information dissemination platforms. Future research directions may explore further optimization of ensemble techniques and integration of advanced deep learning architectures for enhanced performance.

## VI. FUTURE WORK

One promising avenue for future research in fake news detection using machine learning, deep learning, and natural language processing (NLP) techniques with the LIAR dataset is to delve into more sophisticated ensemble methods. These methods could involve combining multiple models from various domains, such as traditional machine learning algorithms and deep neural networks, to create a more robust and accurate detection system. Additionally, exploring advanced NLP techniques like transformer-based models such as BERT or GPT could enhance the understanding of textual data and improve the model's ability to discern fake news. Moreover, investigating techniques for handling imbalanced datasets, as the LIAR dataset contains a mix of true and false statements, would be beneficial for refining the model's performance. Finally, integrating domain-specific knowledge, such as fact-checking information or contextual clues, could further enhance the model's ability to distinguish between genuine and fake news articles.

## REFERENCES

[1] Singh and S. Patidar, "A Survey on Fake News Detection Using Machine Learning," 2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India, 2022, pp. 327-331, doi: 10.1109/ICAC3N56670.2022.10074450.

[2] A.Dnyandeo Waghmare and G. Kumar Patnaik, "Social Media Fake News Detection using mNB in Blockchain," 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2022, pp. 1198-1204, doi: 10.1109/ICSCDS53736.2022.9760840.

[3] S. Mondal, P. Srinivasan and M. E. Ahammed, "Fake News Detection: A Comparative Study of Machine Learning Techniques," 2023 International Conference on Computational Intelligence, Networks and Security (ICCINS), Mylavaram, India, 2023, pp. 1-6, doi: 10.1109/ICCINS58907.2023.10450148.

[4] A. Narkhede, D. Patharkar, N. Chavan, R. Agrawal and C. Dhule, "Fake News Detection Using Machine Learning Algorithm," 2023 International Conference on Communication, Security and Artificial Intelligence (ICCSAI), Greater Noida, India, 2023, pp. 166-170, doi: 10.1109/ICCSAI59793.2023.10421153.

[5] V. S. Pendyala and F. S. A. Tabatabaii, "Spectral analysis perspective of why misinformation containment is still an unsolved problem," 2023 IEEE Conference on Artificial Intelligence (CAI), Santa Clara, CA, USA, 2023, pp. 210-213, doi: 10.1109/CAI54212.2023.00099.

[6] H. Gao, "Spam Sorting Based on Naive Bayes Algorithm," 2023 International Conference on Blockchain Technology and Applications (ICBTA), Beijing, China, 2023, pp. 81-84, doi: 10.1109/ICBTA60381.2023.00023.

[7] S. Warjri, P. Pakray, S. A. Lyngdoh and A. K. Maji, "Fake news detection using social media data for Khasi language," 2023 International Conference on Intelligent Systems, Advanced Computing and Communication (ISACC), Silchar, India, 2023, pp. 1-6, doi: 10.1109/ISACC56298.2023.10083518.

[8] R. Llugsi, "A Hyperparameter Optimization Strategy to Boost Fake News Detection," 2023 IEEE Seventh Ecuador Technical Chapters Meeting (ECTM), Ambato, Ecuador, 2023, pp. 1-6, doi: 10.1109/ETCM58927.2023.10309046.

[9] P. Agarwal, S. Reddivari and K. Reddivari, "Fake News Detection: An Investigation based on Machine Learning," 2022 IEEE 23rd International Conference on Information Reuse and Integration for Data Science (IRI), San Diego, CA, USA, 2022, pp. 61-62, doi: 10.1109/IRI54793.2022.00025.

[10] A. E. Qasem and M. Sajid, "Exploring the i of N-grams with BOW and TF-IDF Representations on Detecting Fake News," 2022 International Conference on Data Analytics for Business and Industry (ICDABI), Sakhir, Bahrain, 2022, pp. 741-746, doi: 10.1109/ICDABI56818.2022.10041537.

[11] P. Santhiya, S. Kavitha, T. Aravindh, S. Archana and A. V. Praveen, "Fake News Detection Using Machine Learning," 2023 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2023, pp. 1-8, doi: 10.1109/ICCCI56745.2023.10128339.

[12] M. Jung, J. Choi and J. Jo, "Projection Ensemble: Visualizing the Robust Structures of Multidimensional Projections," 2023 IEEE Visualization and Visual Analytics (VIS), Melbourne, Australia, 2023, pp. 46-50, doi: 10.1109/VIS54172.2023.00018.

[13] A. Iqbal, M. A. Rauf, M. Zubair and T. Younis, "An Efficient Ensemble approach for Fake Reviews Detection," 2023 3rd International Conference on Artificial Intelligence (ICAI), Islamabad, Pakistan, 2023, pp. 70-75, doi: 10.1109/ICAI58407.2023.10136652.

[14] T. A. Roshinta, Hartatik, E. K. Fauziyah, I. F. Dinata, N. Firdaus and F. Y. A'la, "A Comparison of Text Classification Methods: Towards Fake News Detection for Indonesian Websites," 2022 1st International Conference on Smart Technology, Applied Informatics, and Engineering (APICS), Surakarta, Indonesia, 2022, pp. 59-64, doi: 10.1109/APICS56469.2022.9918702.

[15] N. L. S. R. Krishna and M. Adimoolam, "Fake News Detection system using Decision Tree algorithm and compare textual property with Support Vector Machine algorithm," 2022 International Conference on Business Analytics for Technology and Security (ICBATS), Dubai, United Arab Emirates, 2022, pp. 1-6, doi: 10.1109/ICBATS54253.2022.9758999.

[16] A. A. Kulkarni, P. Rakshith Shenoy, J. Baradia, R. Sharma and A. Arya, "Machine Learning Techniques for Fake News Detection in Low-Resource Hindi Language: A Comparative Study," 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, 2023, pp. 1-7, doi: 10.1109/ICCCNT56998.2023.10307399.

[17] S.Satheesh Kumar, M.Karthick, An Secured Data Transmission in MANET Networks with Optimizing Link State Routing Protocol Using ACO-CBRP Protocols, IEEE Access, 2018.

[18] Karthik.S, Karthick.M., Karthikeyan.N, Kannan.S, A multi-Mobile Agent and optimal itinerary planning-based data aggregation in Wireless Sensor Networks, Computer Communications 184 (2022) 24–35, https://doi.org/10.1016/j.comcom.2021.11.019

[19] Azath Mubarakali, Salomi Samsudeen, Ahmed Alkhayyat, Badria Sulaiman Alfurhood, D. Haritha, Deevi Radha Rani, M. Karthick, Optimized flexible network architecture creation against 5G communication-based IoT using information-centric wireless computing, Wireless Networks,2023 https://doi.org/10.1007/s11276-023-03531-1

[20] M. Karthick, Dinesh Jackson Samuel, B. Prakash, P. Sathyaprakash, Nandhini Daruvuri, Mohammed Hasan Ali, R.S. Aiswarya, Real-time MRI lungs images revealing using Hybrid feed forward Deep Neural Network and Convolutional Neural Network, Intelligent Data Analysis 27 (2023) S95–S114, DOI 10.3233/IDA-237436.

[21] Karthick.M, Chandru Vignesh.C, Alfred Daniel.J, Sivaparthipan.C.B, An Efficient Multi-mobile Agent Based Data Aggregation in Wireless Sensor Networks Based on HSSO Route Planning, Ad Hoc & Sensor Wireless Networks, Vol. 57, pp. 187–207, DOI: 10.32908/ahswn.v57.10319.

[22] Karthick.M, Salomi Samsudeen, Likewin Thomas, Priya Darsini.V, Prabaakaran.K, Cybersecurity Warning System Using Diluted Convolutional Neural Network Framework for IOT Attack Prevention, International Journal of Intelligent Engineering and Systems, Vol.17, No.1, 2024, DOI: 10.22266/ijies2024.0229.66