2024 International Conference on Advances in Computing, Communication,
Electrical, and Smart Systems (iCACCESS), 8-9 March, Dhaka, Bangladesh

Stroke Detection Through Ensemble Learning: A Stacking Approach

Jeba Faria
Dept. of Electronics and Communication Engineering
Khulna University of Engineering & Technology
Khulna-9203, Bangladesh
jebafaria9876@gmail.com

Shah Muhammad Azmat Ullah
Dept. of Electronics and Communication Engineering
Khulna University of Engineering & Technology
Khulna-9203, Bangladesh
azmat@ece.kuet.ac.bd

Md. Rabiul Hasan
Dept. of Electronics and Communication Engineering
Khulna University of Engineering & Technology
Khulna-9203, Bangladesh
mdrabiulhasan7890@gmail.com

Abstract— **A stroke is a health condition that occurs when there is an interruption in the normal blood flow to the brain. Brain cells eventually die due to the lack of blood supply. As per the WHO, it ranks as the fundamental cause of both fatalities and disabilities. Timely identification and acknowledgment of symptoms can swift stroke treatment, leading to enhanced health outcomes. In this work we developed different machine learning models including stacking. The selected models are Random Forest (RF), Decision Tree (DT), Logistic Regression (LR), K-Nearest Neighbors (KNN), and Support Vector Machine (SVM). The base models used for stacking are DT, LR and KNN and for meta model LR is used. The dataset was preprocessed by deleting unnecessary columns, dropping null values, removing outliers and by encoding the categorical values using One Hot Encoding. Oversampling technique is used to balance the dataset as it was highly unbalanced. After evaluation the findings reveal that the proposed stacking ensemble approach performs better than the single models, achieving an accuracy rate of nearly 99%.**

Keywords— ***stroke detection, oversampling, machine learning, ensemble, stacking.***

I. Introduction

Stroke is a significant global health concern, with profound consequences for individuals and society as a whole, resulting from disruptions of blood flow into the brain. This condition stands as a leading cause of both mortality and disability. Its consequences extend across various aspects of life, affecting the body's structural and functional integrity. Stroke survivors often encounter challenges in performing even the most basic activities of daily living (ADLs) [1]. Stroke affects around 800,000 Americans every year, with around 600,000 of these cases being first-time strokes. Surprisingly, the incidence of a first stroke increases the risk of future strokes. This risk is most pronounced immediately after the initial episode and gradually lessens over time. Notably, 25% of patients who recover from their first stroke will have another within five years, whereas approximately 3% will have a second stroke within 30 days[2] .

The accessibility of clinical data stored in electronic health records, known as EHRs, has facilitated the utilization of machine learning (ML) methods for identifying various medical conditions, including stroke, even in its early stages. By early detection of stroke, long term disability and mortality related with stroke can be prevented. Recently, machine learning techniques have been efficiently predicting risk of stroke. Stacked ensemble techniques in machine learning involve merging the outcomes of multiple classification models and have demonstrated superior performance compared to individual models. Previous studies [3]–[10] have effectively employed the machine learning including stacked ensemble approach to aid medical decision-making, particularly regarding stroke detection. Thus, the objective of this study is to employ a stacked ensemble learning method for early-stage stroke detection using clinical data obtained from common health records.

In this work, A publicly available dataset was utilized for detecting strokes. The dataset was at first preprocessed by deleting unnecessary columns, dropping null values, detecting and removing outlier's and converting the categorical values to numerical values using One Hot Encoding. The dataset was highly unbalanced, so we used oversampling to balance the data. Then various models were selected and evaluated on the processed dataset. The primary focus of this study is a stacking method, where three different models were used as base models and one as meta model to enhance the overall performance.

The paper is structured as follows: Section II presents a summary of the literature review. Section III outlines the proposed Stacked Ensemble Model for Stroke Detection, covering dataset descriptions, data preprocessing, existing model description and the proposed model's design used for effective evaluation. Section IV discusses the performance evaluation results for single model and proposed stacked ensemble Model. Finally, Section V offers a summary, conclusion, and discussion of potential avenues for future research.

II. Literature Review

Multiple research studies have utilized machine learning models, resulting in significant findings regarding the detection of stroke. First, In[3], the authors introduced a robust ensemble model for accurate stroke detection using RXLM, which combines XGBoost, Random Forest (RF), and LightGBM . They employed Random Search Optimization to fine-tune the model's parameters and then employed parameter stacking. Their RXLM model achieved an accuracy of 96.34%. Then, in [4] the authors developed and evaluated various models including Logistic regression, stochastic gradient descent (SGD), K-NN, decision trees, random forests, multi-layer perception, and naive Bayes. They further explored ensemble techniques, specifically majority voting and stacking method. The primary focus was to use stacking technique, where the stacking ensemble consists of J48, naive Bayes, random forests, and RepTree as base classifiers. The

979-8-3503-5028-9/24/$31.00 ©2024 IEEE

2024 International Conference on Advances in Computing, Communication, Electrical, and Smart Systems (iCACCESS) | 979-8-3503-5028-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICACCESS61735.2024.10499584

predictions from these base classifiers were utilized to train a logistic regression meta-classifier. The stacking model was most efficient with an accuracy of 98%. Moreover, in[5], the authors explored diverse factors and employed several ML algorithms, for accurate stroke prediction. They also created a user-friendly HTML page with a Flask application, enabling users to input parameters and obtain results. Notably, the NBC(Naïve Bayes Classification) algorithm performed the best with an 82% accuracy rate. Furthermore, the study suggests future extractions involving Neural Networks for model training and the collection of image datasets. Similarly, in[6] several physiological measurements and machine learning techniques, including the Voting Classifier, Logistic Regression (LR), Decision Tree (DT) Classification, and Random Forest (RF) Classification, were utilized to create four separate predictive models. The Random Forest algorithm achieved an accuracy of around 96 percent for brain stroke forecasting. A future scope was suggested where broadening the dataset and exploring advanced machine learning techniques like SVM, AdaBoost and Bagging will enhance the framework model. Research paper[7] shows a comparative performance between six well known classifiers including XGBoost, Naïve Bayes, KNN, Logistic Regression (LR), Random Forest (RF) Classification, and SVM classifiers before and after data leakage. Two interpretable techniques, namely LIME and SHAP were studied for explaining model decision-making. The Random Forest classifier, known for its ensemble learning approach by combining multiple decision trees, outperformed the other classifiers in their experimentation with an accuracy of 90.36%. S. Peñafiel et al[8] introduced an interpretable classifier which was based on the Dempster-Shafer Theory(DST) of plausibility. which not just outperformed other methods but additionally provided insights into the most likely factors contributing to stroke risk. This study introduced the Depmster-Shafer Using Gradient Descent Classifier (DSGD) model, which incorporates the principles of the Dempster-Shafer Theory (DST) in its calculations. The authors compared their proposed DSGD model with some other models where DSGD performed the best with an accuracy of 85.4%. Furthermore, in[9] the author analyzed eleven models including AdaBoost, SVM, Gradient Boosting, Random Forest, Nearest Centroid, Multi-Layer Perception, Voting Classifier, Decision Tree, KNN, and Naïve Bayes to achieve more accurate stroke diagnosis with imbalanced data. This research resulted in the creation of a mobile app and web page for real-time risk assessment. Among the eleven models, the SVM and RF models demonstrated the highest accuracy rates, achieving 99.99% and 99.87%, respectively. And finally, in[10], the authors compared their stroke prediction approach with alternative methods. They collected and preprocessed data, performed feature selection using the C4.5 algorithm with Decision Trees, applied Principal Component Analysis (PCA) for dimensionality reduction, and employed Artificial Neural Networks (ANN) for classification. Their proposed method exhibited superior accuracy when compared to two other methods.

Hence, this paper proposes a Stroke Detection framework that incorporates various data preprocessing methods and a stacked ensemble learning approach for detecting stroke. The projected framework will improve overall model performance and enable individuals to detect the early stages of hypertension.

III. System Methodology

The primary objective of this study is to develop a system capable of identifying strokes. In pursuit of this goal, we devised two distinct approaches. The first approach involves using the results from each separate model, while the second approach involves combining these models into an ensemble. To achieve accurate stroke detection, we employed different machine learning models such as SVM, Decision Tree, Random Forest, KNN and Logistic Regression. Fig. 1 illustrates an overview of the proposed methodology, which encompasses data collection, data preparation, the utilization of individual models, and the creation of a stacked ensemble model.

A. Dataset

Our research relied on a publicly available dataset sourced from Kaggle [11]. This dataset comprises 11 clinical attributes and contains a total of 5110 entries. It's worth noting that the dataset exhibited an imbalance, with just 249 entries showing a positive stroke occurrence, while the majority, 4861 entries, had a negative stroke occurrence.

B. Data Preprocessing

1) Removing Unnecessary Columns

The columns that are not related to the target feature need to be deleted. In this work the “id” column has no relevance with stroke. So, this column has been deleted.

2) Dropping Null Values

There was total 201 null values identified in the BMI feature which need to be handled. After counting these null values, they were dropped for proper functioning of the machine learning algorithm. Dropping null values may lead to valuable information loss.

3) Detecting and Removing Outliners

Outliers are data points that are significantly different from majority of the data from dataset. They can result from

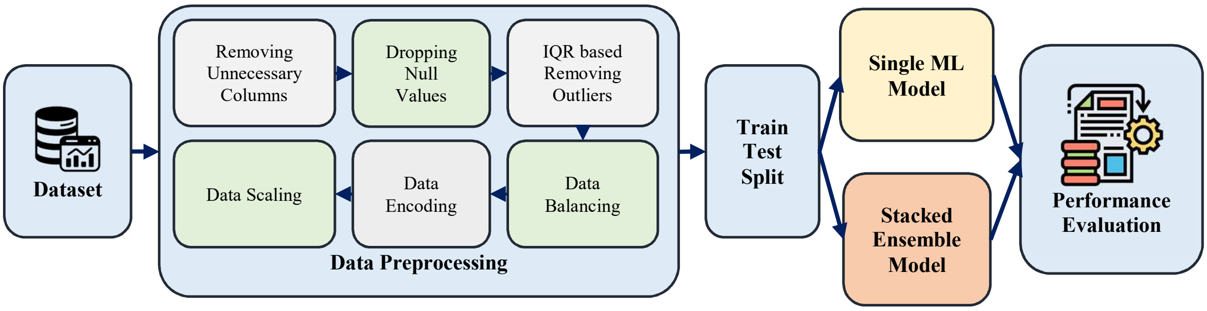


Fig. 1. Flow Diagram of Proposed Stroke Detection Technique.

measurement or execution errors. In this work a total number of 649 outliers were detected by using the Interquartile Range (IQR) method and then they were removed from the dataset. Before removing the outliers there were total 4909 samples and after removing outliers there were total 4260 samples.

4) Data Balancing

Imbalanced data is a prevalent issue in classification tasks where one target class label is significantly greater than the other class label. In this work the data was highly imbalanced with Only 136 samples indicating a stroke, while a vast majority of 4,124 samples exhibited no signs of stroke. In this study to match the number of samples between majority class (No Stroke) and minority class (Stroke) oversampling method was used.

5) Data Encoding

Data encoding is a way of transforming the categorical values to numerical values as many machine learning algorithms require numerical inputs to perform calculations and make predictions. In this work one Hot Encoding has been used for encoding creating binary columns for each category.

6) Data Scaling

Data scaling is a process which is used to transform the features or variables to a similar scale. It ensures equal contribution of all features to the learning algorithm. In this study StandardScaler has been used for scaling.

C. Model Description

1) Logistic Regression.

Logistic Regression is a simple and interpretable supervised machine learning model which is generally employed for binary classification problem, though it can be extended to solve multiclass classification problem. It is used when the data is linearly separable and the outcome is binary or dichotomous in nature[12]. The core of this model relies on the sigmoid function, creating its characteristic "S"-shaped curve. The output of this model may be yes or no, 0 or 1, true or false, etc. In this work the output belongs either to class “Stroke” or class “No Stroke”.

2) Decision Tree

Decision Tree (DT) is a widely-used Machine Learning model applicable to both regression and classification tasks. It is known for its interpretability which makes it easier to understand the decision making. It forms a hierarchical structure where data is divided into branches, leading to predictions at the leaf nodes. In this model, internal nodes represent features, while leaf nodes correspond to classes. There are two main components of a decision tree: decision nodes, where data is split, and leaf nodes, where predictions are made. If the tree is deep it may lead to overfitting.

3) Random Forest

Random Forest (RF) is an algorithm used for supervised machine learning and widely employed for both regression and classification tasks. It operates as an ensemble method, consisting of numerous separate decision trees, each trained on a randomly selected subset of the data. For final predictions, a voting mechanism is applied. In this study RF achieved the second-best accuracy. One of the notable strengths of the Random Forest algorithm is its resistance to overfitting, a common concern in machine learning. If there are enough trees present in the forest, the classifier will avoid overfitting the model.

4) KNN

K-Nearest Neighbors is a distance-based classifier used in supervised learning techniques. It is considered as non-parametric algorithm because it doesn't rely on any assumptions about the underlying data distribution. KNN is a "lazy" algorithm, which means it doesn't immediately train on the dataset. Instead, it stores the dataset and performs actions on it during classification. The choice of K is very critical, as small value of K leads to overfitting and large value of K leads to underfitting. KNN is useful when the dataset in not very large. Its flexibility in handling complex data patterns makes it a valuable tool in various machine learning applications.

5) Support Vector Machine

SVM is a type of supervised Machine Learning algorithm utilized tasks in both classification and regression, though especially well-suited for classification tasks. This classification identifies the optimal hyperplane that separates the dataset into two classes [13]. In this work SVM achieved an accuracy of 89%.

D. Ensemble Learning

In this research, we utilized stacking, an ensemble learning technique that combines diverse classifiers within a meta-classifier. Our approach involves training base models (Decision Tree, Logistic Regression, and KNN) on the training data and then using their predictions to train a logistic regression meta-classifier. Fig. 2 illustrates the structure of the proposed model. Here, the stacked ensemble learning was implemented using Scikit-learn V0.20.2 python libraries.

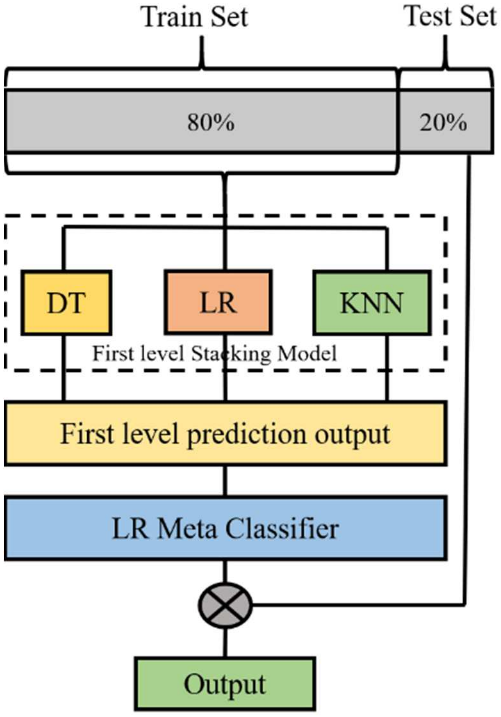
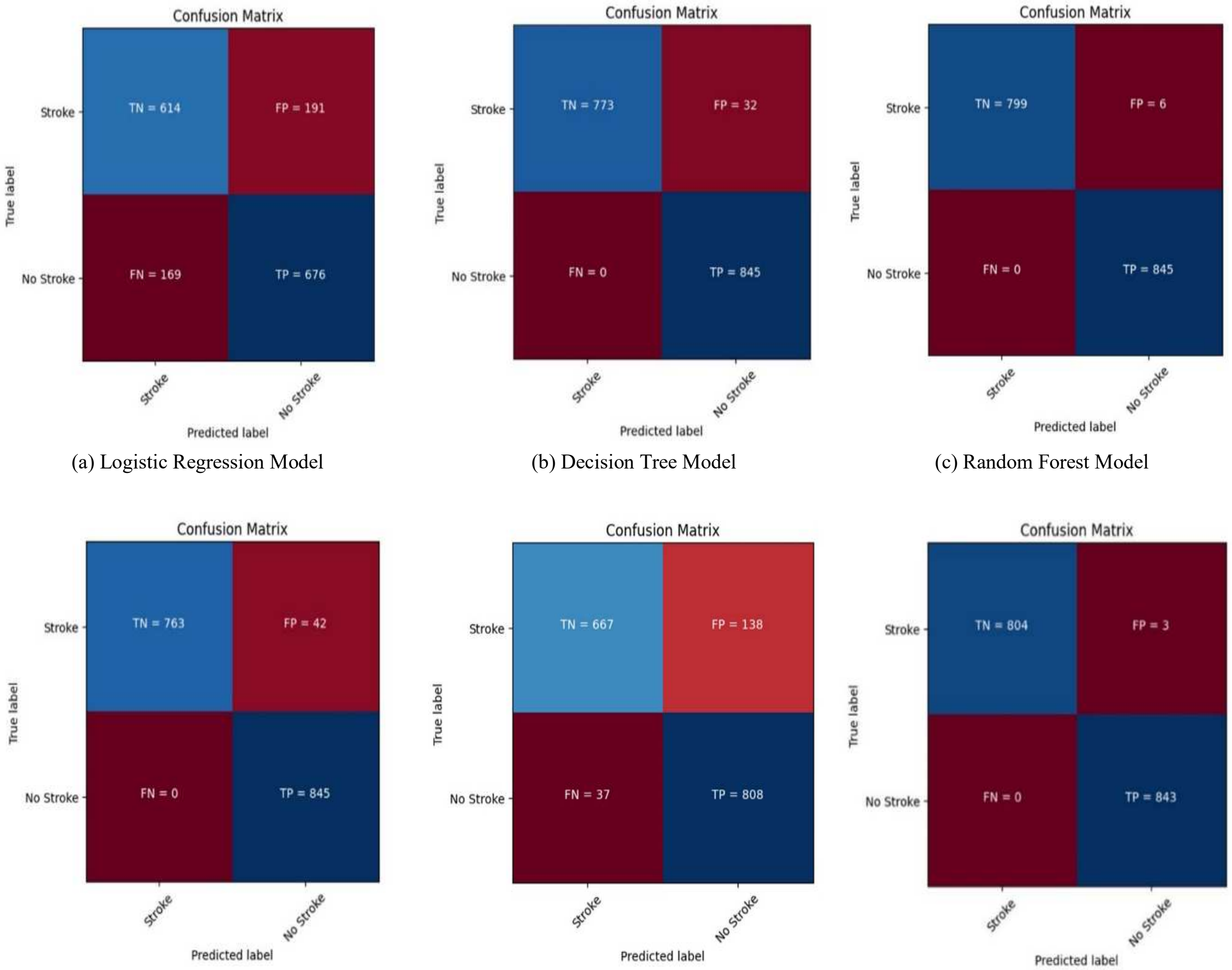


Fig. 2. Proposed Stacked Ensemble Model for Stroke Detection.



(a) Logistic Regression Model

(b) Decision Tree Model

(c) Random Forest Model

(d) KNN Model

(e) SVM Model

(f) Proposed Stacked Ensemble Model.

Fig. 3. Confusion Matrix for Different Models

IV. Results and Discussion

In this study, we initially assessed the performance of different individual machine learning models by employing a confusion matrix. Fig. 3 and Table I shows the performance of Decision Tree, SVM, Logistic Regression, Random Forest and KNN models and proposed stacked ensemble model.

A. Single Model Evaluation

1) Logistic Regression.

Fig. 3(a) illustrates the performance of Logistic Regression by using confusion matrix. The accuracy achieved with this algorithm is approximately 77%. Additional metrics evaluating this model's efficiency include precision (76%), recall (79%), and F-1 score (78%) for the "stroke" case. In this study Logistic Regression performed the worst out of all the models.

2) Decision Tree

The confusion matrix of Decision Tree is shown in the Fig. 3(b), which achieved an accuracy of approximately 98% The confusion matrix reveals that the model had zero false negatives, meaning it correctly identified all instances of the "No Stroke" class. The precision, recall and F-1 score for the "stroke" case are 97%, 100% and 98% respectively.

3) Random Forest.

Fig. 3(c) depicts the confusion matrix of Random Forest (RF). From the confusion matrix it is seen that there are only 6 erroneous predictions. In this work this model achieved the second-best accuracy out of all the models which is 99.51%. The recall, F1-score and precision are also very high which are 100%, 100% and 99% respectively for stroke class.

4) KNN.

The confusion matrix from Fig. 3(d) shows a high performance of KNN model. The confusion matrix reveals that there were a total of 1608 accurate guesses with only 42 erroneous predictions. This model achieved a high accuracy of around 97%. For the stroke class precision and F1-score reached 95% and 97%, respectively, while recall was 100%.

5) Support Vector Machine.

The confusion matrix in Fig. 3(e) reveals that, the accuracy achieved by SVM model is around 89%. The precision, recall, and F1-score for the "stroke" class are 85%, 94%, and 89%, respectively. The confusion matrix indicates a total of 1475 accurate predictions with 175 erroneous ones.

TABLE I. MODEL PERFORMANCE EVALUATION.

Model	Accuracy (%)	Precision (%)		Recall (%)		F1-score (%)	
		0	1	0	1	0	1
Logistic Regression	77	78	76	75	79	77	78
Decision Tree	98	100	97	97	100	98	99
Random Forest	99.51	100	99	99	100	100	100
KNN	97	100	95	94	100	97	97
SVM	89	94	85	83	94	88	89
Proposed Stacked Ensemble Model	99.63	100	99	99	100	100	100

Here, in table I, no stroke=0 and stroke =1

B. Ensemble Model Evaluation

The confusion matrix depicted in Fig. 3(f) reveal that the stacking ensemble model outperformed all other models, achieving an accuracy rate of 99.63%. For the stroke class, the precision is 99%, while the recall and F1-score are both 100%. Remarkably, only 6 incorrect predictions were observed, as shown in the confusion matrix.

Table II presents a comparison between the proposed model and with the outputs from previous papers. Our findings indicate that the proposed model surpassed all other models, outperforming the results from both previous studies across accuracy.

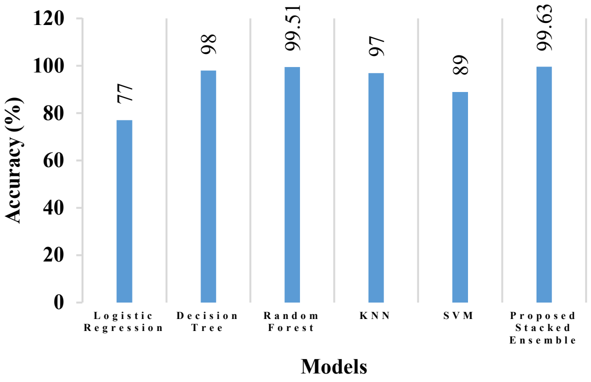


Fig. 4. Accuracy Comparison of Single and Ensemble Models.

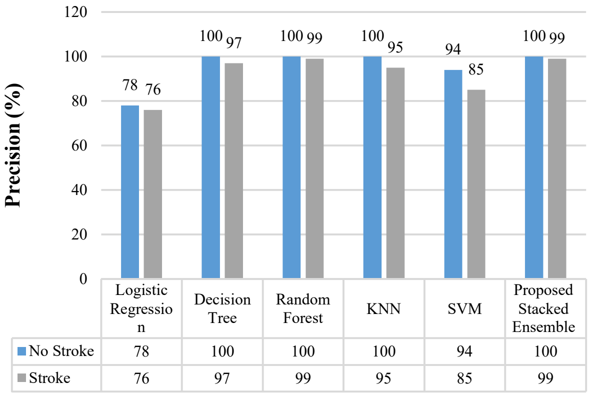


Fig. 5. Precision Comparison of Single and Ensemble Models.

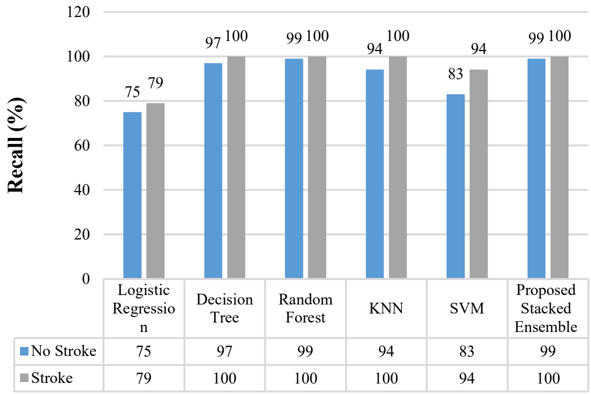


Fig. 6. Recall Comparison of Single and Ensemble Models.

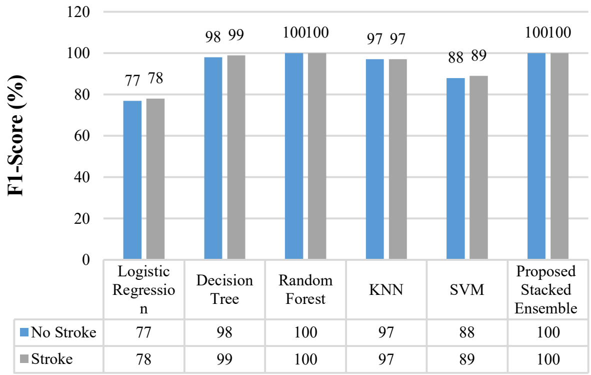


Fig. 7. F1-Score Comparison of Single and Ensemble Models.

TABLE II. PERFORMANCE COMPARISON WITH EXISTING MODELS

Authors	Methodology	Outcomes
Alruily et al.[3]	Random Forest (RF), XGBoost, and LightGBM	Accuracy 96.34%
Dritsas et al.[4]	Decision Tree, Random Forests, SGD, Logistic Regression, KNN, and Multi-Layer Perception.	Accuracy 98.9%
Sailasya et al.[5]	Random Forest, Decision Tree, KNN, SVM, Naïve Bayes, and Logistic Regression Models.	Accuracy 82%
Mridha et al.[7]	Random Forest, SVM, Logistic Regression, Naïve Bayes, KNN, and XGB.	Accuracy 90.36%
Proposed Model	Stacking ensemble of Logistic Regression, Decision Tree and KNN with LR as meta level	**Accuracy 99.63%**

The comparison between the chosen individual models and the stacking ensemble method in terms of performance metrics presented in Fig. 4, 5, 6, 7. These figures clearly illustrate that the stacking method surpasses all other approaches in performance.

V. CONCLUSION

In this study we proposed a stacking ensemble approach and compared its performance with traditional single classifiers, namely SVM, Decision Tree(DT), KNN, Logistic Regression(LR) and Random Forest. For our stacking ensemble approach, we selected LR, DT, and KNN as base models, with LR as the Meta model.The output of the base models is taken by the Meta model as its input. We began by collecting a dataset, preprocessing it, selecting various classifiers, evaluating them and finally comparing their performance with our proposed stacking approach.The preprocessing process includes deleting unnecessary columns, dropping null values, detecting and removing outliers, data balancing using oversampling method, data encoding for converting categorical values to numerical values and data scaling by using Standardscaler. The F1-score, recall, precision and accuracy of our proposed stacking method are 100%, 100%, 99%, 99.63% respectively. These findings clearly show that our stacking approach outperforms individual classifiers.

ACKNOWLEDGMENT

The authors would like to thank the Khulna University of Engineering & Technology, Khulna, Bangladesh for different supports.

REFERENCES

[1] T. Elloker and A. J. Rhoda, "The relationship between social support and participation in stroke: a systematic review," African Journal of Disability, vol. 7, no. 1, pp. 1-9, 2018.

[2] M. A. Moskowitz, J. C. Grotta, and W. J. Koroshetz, "The NINDS stroke progress review group final analysis and recommendations," Stroke, vol. 44, no. 8, pp. 2343-2350, 2013.

[3] M. Alruily, S. A. El-Ghany, A. M. Mostafa, M. Ezz, and A. A. El-Aziz, "A-Tuning Ensemble Machine Learning Technique for Cerebral Stroke Prediction," Applied Sciences, vol. 13, no. 8, pp. 5047, 2023.

[4] E. Dritsas and M. Trigka, "Stroke risk prediction with machine learning techniques," Sensors, vol. 22, no. 13, pp. 4670, 2022.

[5] G. Sailasya and G. L. A. Kumari, "Analyzing the performance of stroke prediction using ML classification algorithms," International Journal of Advanced Computer Science and Applications, vol. 12, no. 6, 2021.

[6] T. Tazin, M. N. Alam, N. N. Dola, M. S. Bari, S. Bourouis, and M. Monirujjaman Khan, "Stroke disease detection and prediction using robust learning approaches," Journal of Healthcare Engineering, 2021.

[7] K. Mridha, S. Ghimire, J. Shin, A. Aran, M. M. Uddin, and M. F. Mridha, "Automated Stroke Prediction Using Machine Learning: An Explainable and Exploratory Study with a Web Application for Early Intervention," IEEE Access, 2023.

[8] S. Penafiel, N. Baloian, H. Sanson, and J. A. Pino, "Predicting stroke risk with an interpretable classifier," IEEE Access, vol. 9, pp. 1154-1166, 2020.

[9] N. Biswas, K. M. M. Uddin, S. T. Rikta, and S. K. Dey, "A comparative analysis of machine learning classifiers for stroke prediction: A predictive analytics approach," Healthcare Analytics, vol. 2, pp. 100116, 2022.

[10] M. S. Singh and P. Choudhary, "Stroke prediction using artificial intelligence," in 2017 8th Annual Industrial Automation and Electromechanical Engineering Conference (IEMECON), pp. 158-161, IEEE, August 2017.

[11] Stroke Prediction Dataset. Available online: https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset (Accessed: Dec. 22, 2023).

[12] H. Bonthu, "An Introduction to Logistic Regression," Analytics Vidhya, 2021.

[13] Amendolia, S. R., Cossu, G., Ganadu, M. L., Golosio, B., Masala, G. L., & Mura, G. M. (2003). “A comparative study of k-nearest neighbour, support vector machine and multi-layer perceptron for thalassemia screening”. Chemometrics and Intelligent Laboratory Systems, 69(1-2), 13-20.