IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

AI BASED STOKE PREDICTION USING EMG SIGNALS

[1] Prof. S. N. Bhadane, [2] Reena Mahajan, [3] Apeksha Ahire, [4] Pranjal Nikum, [5] Dipesh Manwani

[1] Associate Professor, [2*3*4*5] Students of Information Technology Department

[1*2*3*4*5] Department Of Information Technology

[1*2*3*4 *5] Pune Vidyarthi Griha's College of Engineering, Nashik, Maharashtra, India

Abstract: Stroke is a leading cause of disability and mortality worldwide, underscoring the critical need for accurate and timely prediction methods. Recent advancements in machine learning techniques offer promising avenues for stroke prediction, leveraging various physiological signals such as electromyography (EMG). In this study, we investigate the efficacy of machine learning algorithms, including Random Forest, XGBoost, K-Nearest Neighbors (KNN), Support Vector Classifier (SVC), and Decision Trees, for stroke prediction using EMG signals. Our findings reveal high accuracies achieved by Random Forest (95.90%) and XGBoost (95.60%), showcasing their potential for early stroke detection. These results underscore the utility of machine learning in healthcare and highlight the significance of early intervention for improving patient outcomes.

Index Terms - EMG Signals, K-Nearest Neighbors (KNN), Support Vector Classifier (SVC), and Decision Trees.

I. INTRODUCTION

Stroke remains a major global health concern, accounting for substantial morbidity, mortality, and healthcare costs. Early detection and intervention are critical for mitigating the devastating effects of stroke, emphasizing the need for accurate prediction models. Conventional stroke risk assessment tools often rely on demographic and clinical factors, which may have limitations in predicting individualized stroke risk. In recent years, the integration of machine learning techniques with physiological signals has emerged as a promising approach for enhancing stroke prediction accuracy.

Electromyography (EMG) signals, which reflect the electrical activity of muscles, offer valuable insights into motor function and neurological disorders. By analyzing EMG signals, machine learning algorithms can detect subtle changes indicative of stroke risk, providing clinicians with actionable insights for timely intervention. In this study, we explore the effectiveness of several machine learning algorithms, including Random Forest, XGBoost, KNN, SVC, and Decision Trees, in predicting stroke based on EMG signals. Our investigation aims to evaluate the performance of these algorithms and identify optimal models for accurate stroke prediction.

Through comprehensive experimentation and evaluation, we aim to contribute to the growing body of literature on machine learning-based stroke prediction. By elucidating the strengths and limitations of different algorithms, our study seeks to inform clinicians, researchers, and healthcare policymakers about the potential of machine learning in improving stroke risk assessment and patient care. Ultimately, our research endeavors to advance the field of stroke prediction, paving the way for personalized and proactive approaches to stroke management and prevention.

II. LITERATURE REVIEW

Electromyography Signal Classification Using Deep Learning (2021)

Author: Mekia Shigute Gaso, Selcuk Cankurt, Abdulhamit Subasi

In this Paper, implemented a deep learning model with L2 regularization and trained it on Electromyography (EMG) data. The data comprises of EMG signals collected from control group, myopathy and ALS patients. Our proposed deep neural network consists of eight layers; five fully connected, two batch normalization and one dropout layers. The data is divided into training and testing sections by subsequently dividing the training data into sub-training and validation sections. Having implemented this model, an accuracy of 99 percent is achieved on the test data set. The model was able to distinguishes the normal cases (control group) from the others at a precision of 100 percent and classify the myopathy and ALS with high accuracy of 97.4 and 98.2 percents, respectively. Thus we believe that, this highly improved classification accuracies will be beneficial for their use in the clinical diagnosis of neuromuscular disorders.

AI-Based Stroke Disease Prediction System Using Real-Time Electromyography Signals (2020)

Jaehak Yu, Sejin Park, Soon-Hyun Kwon, Chee Meng Benjamin Ho, Cheol-Sig Pyo, Hansung Lee.

Stroke is a leading cause of disabilities in adults and the elderly which can result in numerous social or economic difficulties. If left untreated, stroke can lead to death. In most cases, patients with stroke have been observed to have abnormal bio-signals (i.e., ECG). Therefore, if individuals are monitored and have their bio-signals measured and accurately assessed in real-time, they can receive appropriate treatment quickly. However, most diagnosis and prediction systems for stroke are image analysis tools such as CT or MRI, which are expensive and difficult to use for real-time diagnosis. In this paper, we developed a stroke prediction system that detects stroke using real-time bio-signals with artificial intelligence (AI). Both machine learning (Random Forest) and deep learning (Long Short-Term Memory) algorithms were used in our system. EMG (Electromyography) bio-signals were collected in real time from thighs and calves, after which the important features were extracted, and prediction models were developed based on everyday activities. Prediction accuracies of 90.38% for Random Forest and of 98.958% for LSTM were obtained for our proposed system. This system can be considered an alternative, low-cost, real-time diagnosis system that can obtain accurate stroke prediction and can potentially be used for other diseases such as heart disease.

III. AIM AND OBJECTIVE

1. To Extract relevant features from the EMG signals that capture muscle activity patterns, including time-domain and frequency-domain features.
2. To Implement and evaluate multiple machine learning algorithms, including XG-Boost, Random Forest Classifier, K-Nearest Neighbors (KNN), Support Vector Classifier (SVC), and Decision Trees.
3. To Develop and fine-tune these models to predict stroke outcomes
4. To Compare the performance of different algorithms to identify the most effective model for stroke prediction

IV. MOTIVATION

The motivation behind the AI-Driven Stroke Prediction project is rooted in addressing a critical healthcare challenge. Strokes, a leading cause of morbidity and mortality globally, demand swift detection for effective intervention. Traditional methods often fall short in providing timely diagnoses, prompting the need for an innovative solution. Leveraging artificial intelligence and electromyography signals, this project aims to revolutionize early detection methodologies. By synthesizing realistic EMG data and implementing advanced machine learning algorithms, the system endeavors to predict stroke likelihood with high accuracy. The potential impact on patient outcomes is profound, offering the medical community a powerful tool for early intervention and improving the overall quality of stroke care. Ethical considerations, transparency, and user-friendly interfaces underscore the project's commitment to responsible AI implementation in healthcare, aligning with the broader goal of advancing medical technology for societal well-being.

V. METHODOLOGY

Data Collection and Preprocessing: Acquire a diverse dataset of electromyography (EMG) signals from stroke patients and healthy individuals. Ensure the dataset is labeled with corresponding stroke diagnosis status. Preprocess the data by removing noise, filtering, and normalizing to enhance signal quality.

Synthesis of Realistic EMG Data: Generate synthetic EMG data to augment the existing dataset, ensuring a balanced representation of stroke and non-stroke cases. Use generative techniques like Generative Adversarial Networks (GANs) to simulate realistic EMG signals.

Feature Extraction: Extract relevant features from the EMG signals to capture key characteristics indicative of stroke likelihood. Common features may include time-domain statistics, frequency-domain features, and time-frequency representations.

Algorithm Implementation and Training: Implement and train multiple machines learning algorithms, including:

1. XGBoost
2. Random Forest Classifier
3. K-Nearest Neighbors (KNN)
4. Support Vector Classifier (SVC)
5. Decision Trees

Utilize a cross-validation approach to assess model performance and mitigate overfitting.

Hyperparameter Tuning: Perform hyperparameter tuning for each algorithm to optimize their performance. Utilize techniques such as grid search or random search to find the best combination of hyperparameters.

Model Evaluation: Evaluate the performance of each model using appropriate metrics like accuracy, precision, recall, F1-score, and area under the receiver operating characteristic curve (AUC-ROC). Compare the models to identify the most effective algorithm for stroke prediction.

Ethical Considerations and Transparency: Ensure that the data used for training and evaluation is handled with utmost privacy and security, in compliance with applicable regulations and ethical guidelines. Maintain transparency in the data sources, preprocessing steps, and model training process to facilitate reproducibility and trust.

User-Friendly Interface Development: Design and develop a user-friendly interface for healthcare professionals to interact with the AI-driven stroke prediction system. Prioritize ease of use and interpretability of the predictions provided by the system.

VI. Architecture

1. Data Generation:

Synthetic Data: Generate synthetic EMG data that simulates a variety of stroke and non-stroke scenarios. This data will serve as a controlled dataset for model development and testing. Data Labeling: Assign appropriate labels to the synthetic data to indicate the presence or absence of a stroke. Ensure a balanced distribution of stroke and non-stroke samples.

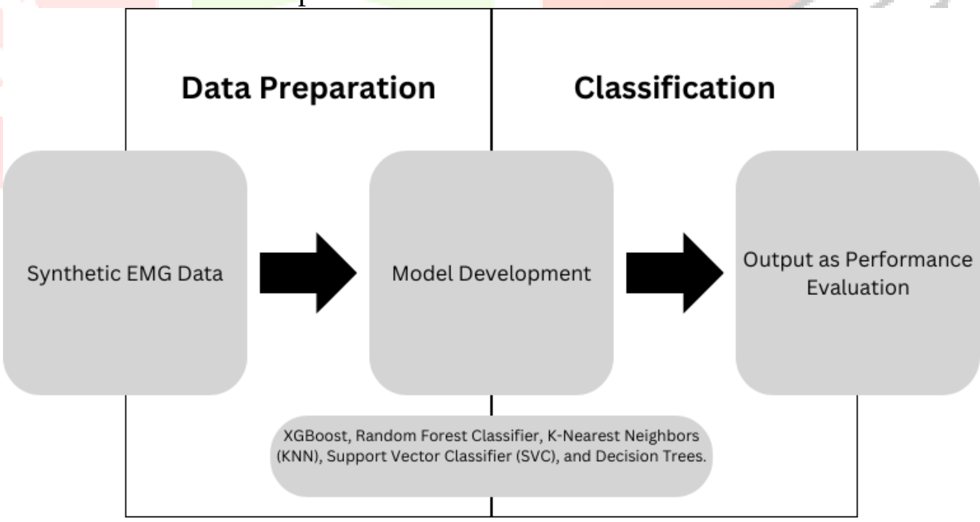


Figure No.1: Architecture of stroke occurrence

2. Data Preprocessing Noise Injection:

Introduce realistic noise and artifacts into the synthetic EMG data to mimic real-world recording conditions. Feature Extraction: Extract relevant features from the synthetic EMG signals, simulating the process used with real EMG data.

3. Model Development:

Algorithm Selection: Choose machine learning algorithms for stroke prediction, including XGBoost, Random Forest Classifier, K-Nearest Neighbors (KNN), Support Vector Classifier (SVC), and Decision Trees. Synthetic Data Split: Divide the synthetic data into training, validation, and test sets for model development and evaluation. Hyperparameter Tuning: Optimize model hyperparameters using techniques like grid search or random search.

4. **Performance Evaluation: Metrics:**

Evaluate model performance using metrics such as accuracy, precision, recall, F1-score, and AUC-ROC. Implement these metrics to measure the quality of predictions. Cross-Validation: Implement k-fold cross-validation to ensure robust model assessment

VII. MODELING AND ANLYSIS

Algorithms

1. **XGBoost**

XGBoost operates in function space using the Newton-Raphson method, in contrast to gradient boosting which employs gradientdescent. It utilizes a second-order Taylor approximation in the loss function to establish this connection to the Newton-Raphson approach. XGBoost is an extremely efficient and scalable implementation of the boosting algorithm, offering performance on par with other leading machine learning algorithms in most scenarios.

XGBoost, short for Extreme Gradient Boosting, is a distributed gradient boosted decision tree (GBDT) machine learning library known for its scalability. It supports parallel tree boosting and is the foremost library for regression, classification, and ranking problems.

To comprehend XGBoost, it's crucial to first grasp the foundational machine learning concepts and algorithms it builds upon: supervised machine learning, decision trees, ensemble learning, and gradient boosting. Supervised machine learning involves training a model to identify patterns in a labeled dataset's features and then using the trained model to predict labels on a new dataset.

2. **Random forests**

Random forests are an ensemble learning method that combines multiple decision trees into a collective model, resulting in more accurate and stable predictions. This technique is based on the premise that a large assembly of trees, acting as a committee (forming a strong learner), will outperform a single constituent tree (which is a weak learner). This is akin to the statistical requirement of having a sufficiently large sample size for meaningful results.

While individual trees within a random forest may make errors, as long as they are not making entirely random predictions, their collective output will approximate the underlying data. Conversely, reducing k leads to an increase in variance while the bias remains consistent throughout the process. This process can be expressed as follows: random forest = dt (base learner) + bagging (row sampling with replacement) + feature bagging (column sampling) + aggregation (mean/median, majority vote).

3. **K-Nearest Neighbours (KNN)**

The K-Nearest Neighbors (K-NN) algorithm is a straightforward machine learning technique based on supervised learning. It operates on the assumption of similarity between new and existing data points, placing the new data point into the category most akin to the available ones. The K-NN algorithm retains all the accessible data and classifies a new data point based on similarity. This implies that when new data emerges, it can be effectively categorized using the K-NN algorithm.

K-NN can be employed for both regression and classification tasks, although it is predominantly used for classification. It is classified as a non-parametric algorithm, meaning it doesn't make any assumptions about the underlying data. Additionally, it is often referred to as a lazy learner because it doesn't immediately learn from the training set; instead, it retains the dataset and takes action on it at the time of classification.

The operation of the K-NN algorithm can be elucidated through the following steps:

Step 1: Choose the number K of neighbors.
Step 2: Compute the Euclidean distance for K neighbors.
Step 3: Select the K nearest neighbors based on the calculated Euclidean distance.
Step 4: Among these K neighbors, tally the number of data points in each category.
Step 5: Assign the new data point to the category with the highest neighbor count.
Step 6: Our model is now prepared for use.

4. **SVM**

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes, making it easier to classify new data points in the future. SVM identifies the extreme points, known as support vectors, which play a crucial role in defining the hyperplane. This characteristic gives rise to the name of the algorithm, Support Vector Machine.

5. Decision tree

Decision Tree is a supervised learning technique utilized for both classification and regression problems, although it is primarily favored for classification tasks. It adopts a tree-like structure, where internal nodes represent dataset features, branches depict decision rules, and each leaf node signifies an outcome. Within a Decision Tree, two key nodes are present: the Decision Node and the Leaf Node. Decision nodes facilitate decision-making and have multiple branches, while leaf nodes serve as the ultimate output and do not lead to further branches. The decisions or tests are executed based on the dataset's features. This method provides a graphical representation of potential solutions to a decision-making problem based on given conditions. The term "decision tree" stems from its tree-like progression, starting with a root node that extends into branches, ultimately forming a tree-like framework. To construct a tree, the CART algorithm (Classification and Regression Tree algorithm) is employed.

VIII. Requirements

1. Hardware Requirements

Processor:
Dual-core processor or equivalent (e.g., Intel Core i3 or AMD Ryzen 3)

RAM:
Minimum 4 GB RAM (8 GB or more recommended for better performance, especially when working with large datasets)

Storage:
HDD/SSD with a minimum of 256 GB or more (larger storage capacity is beneficial when handling substantial datasets)

2. Software Requirements

Operating System:
Windows 7 and above or Linux-based system (e.g., Ubuntu 16.04 and above)

Programming Languages and Frameworks:
Python 3.x (Required for data analysis, machine learning, and user interface development)
Relevant Python libraries and frameworks (e.g., NumPy, SciPy, scikit-learn, TensorFlow, Keras).

IX. Applications

a. **Early Stroke Detection:** The system can assist healthcare professionals in identifying individuals at high risk of stroke by analyzing electromyography (EMG) signals. Early detection allows for timely intervention and treatment, potentially reducing the severity and long-term consequences of strokes.
b. **Patient Risk Assessment:** Medical professionals can use the AI system as a supplementary tool for assessing the stroke risk of patients, especially those with risk factors such as hypertension, diabetes, or a history of cardiovascular disease.
c. **Stroke Prevention:** By identifying individuals at risk of stroke, the system can support healthcare providers in recommending preventive measures, lifestyle changes, and medical interventions to reduce the likelihood of stroke occurrence.
d. **Personalized Medicine:** The AI system can contribute to personalized treatment plans by tailoring stroke prevention strategies and therapies based on an individual's specific risk profile, improving the efficacy of medical care.
e. **Clinical Decision Support:** It can serve as a decision support tool in clinical settings, aiding medical professionals in making informed decisions about patient care, including referrals for specialized stroke assessments.

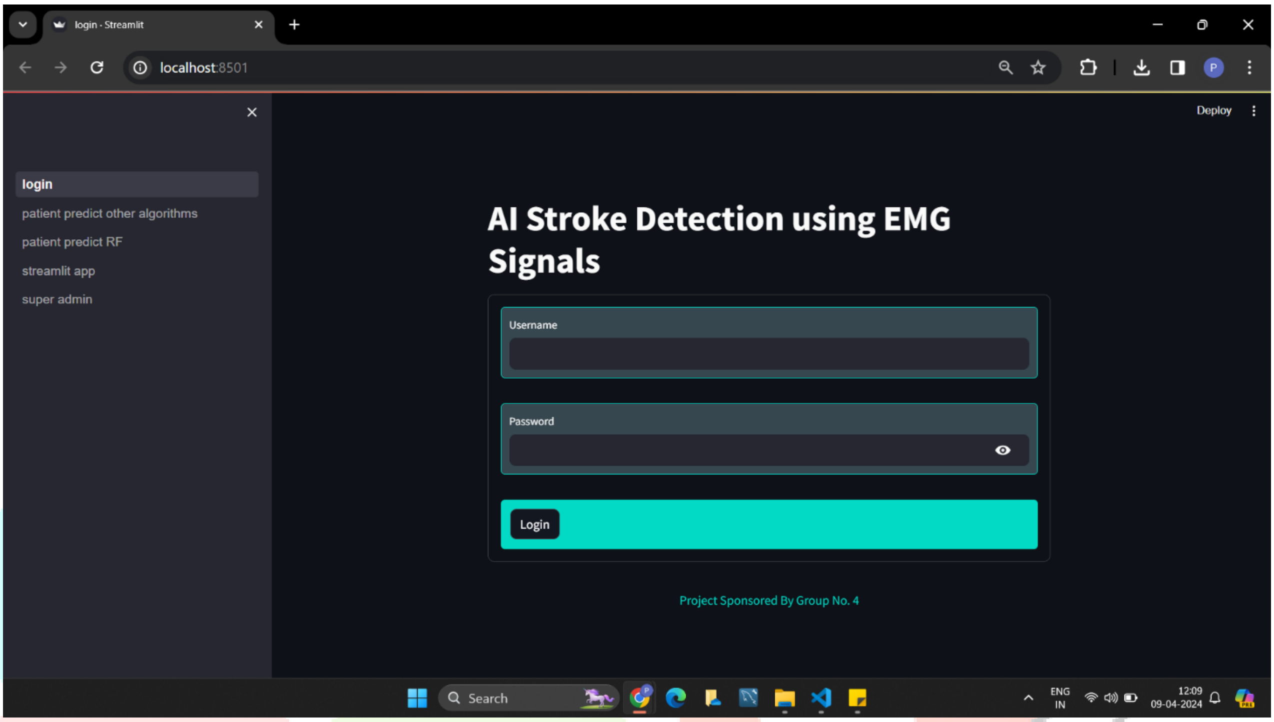
X. RESULTS AND DISCUSSION

The performance of different machine learning algorithms for stroke prediction using EMG signals was evaluated in this study. The results are summarized as follows:

1. Random Forest achieved an accuracy of 95.90%, making it the top-performing model among those evaluated.
2. XGBoost demonstrated competitive performance with an accuracy of 95.60%.

3. K-Nearest Neighbors (KNN) yielded an accuracy of 94.73%, showcasing its effectiveness for stroke prediction.
4. Support Vector Classifier (SVC) achieved an accuracy of 90.99%, exhibiting slightly lower performance compared to Random Forest and XGBoost.
5. Decision Tree, while still showing reasonable accuracy at 86.45%, had the lowest performance among the models tested.

These results indicate the potential of machine learning algorithms, particularly Random Forest and XGBoost, for accurate stroke prediction based on EMG signals. The high accuracies achieved by these models underscore their utility in clinical settings for early detection and intervention.



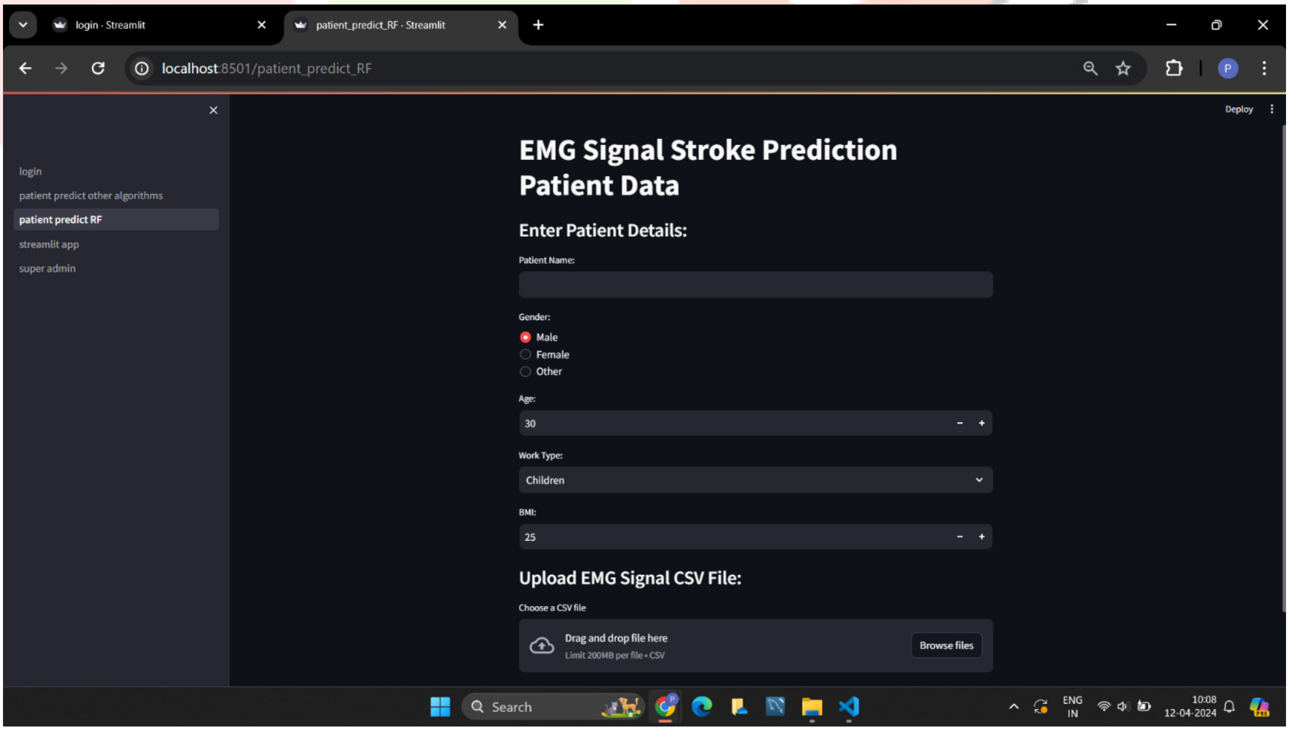


Figure No.2: Login Page

Figure No.3: Predict Random Forest Page

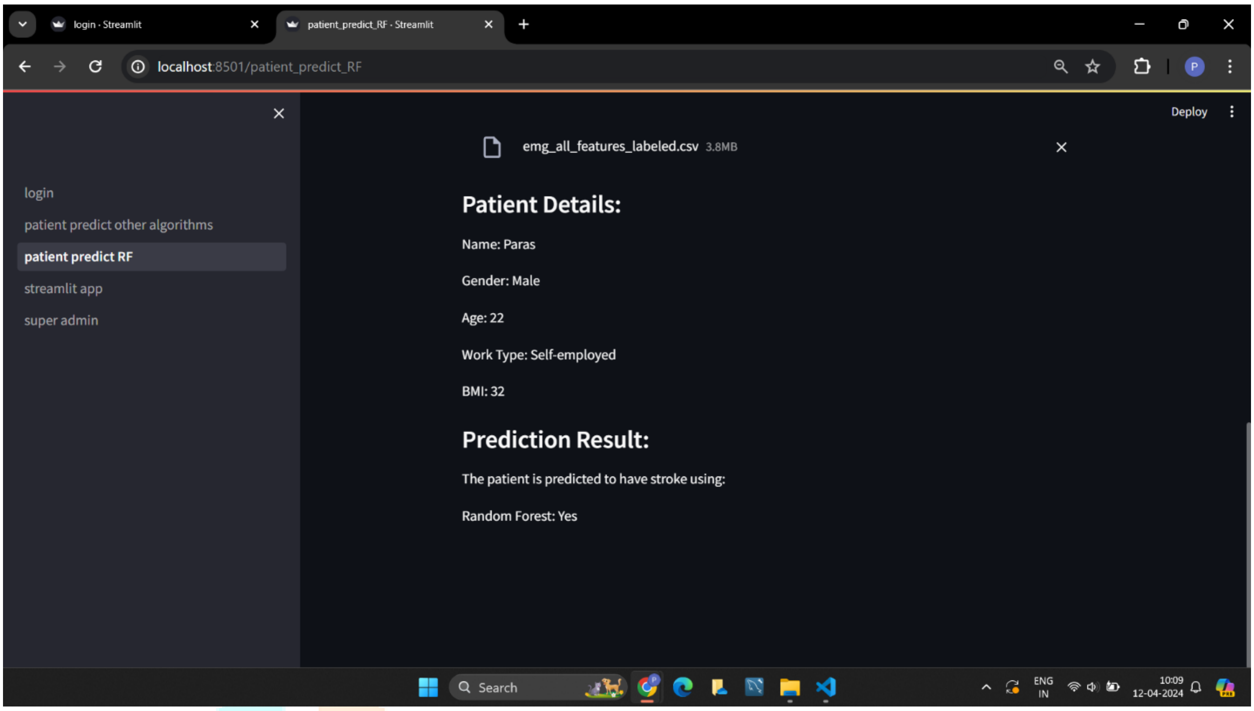
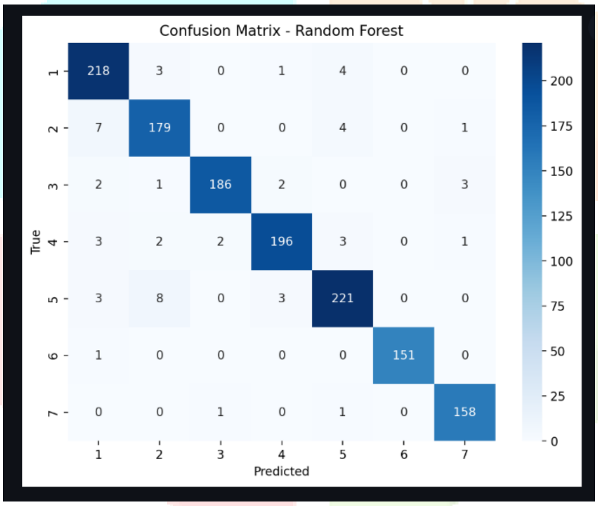


Figure No.4: Predict other algorithm Page



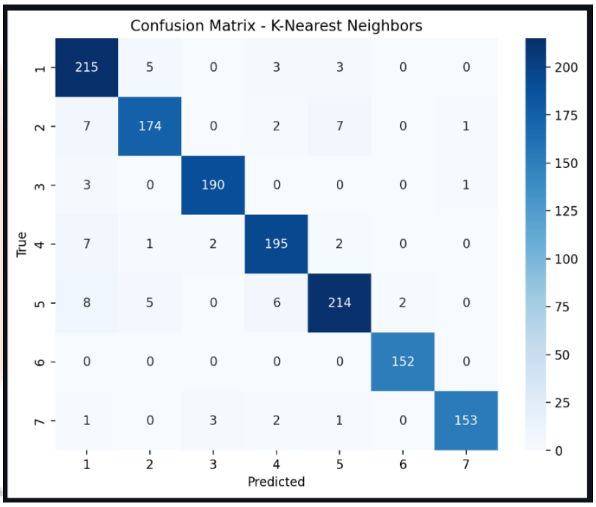
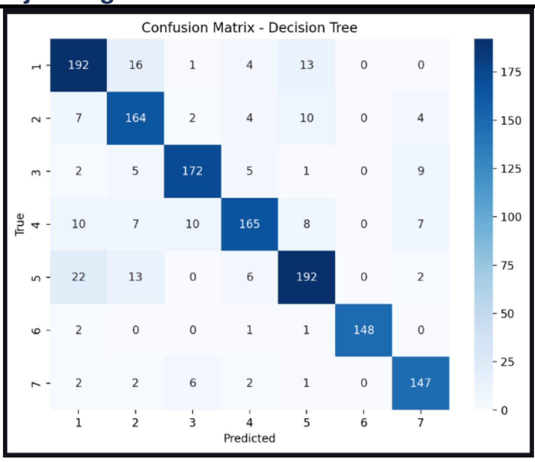


Figure No.5: Confusion Matrix (Random Forest) **Figure No.6: Confusion Matrix (KNN's)**



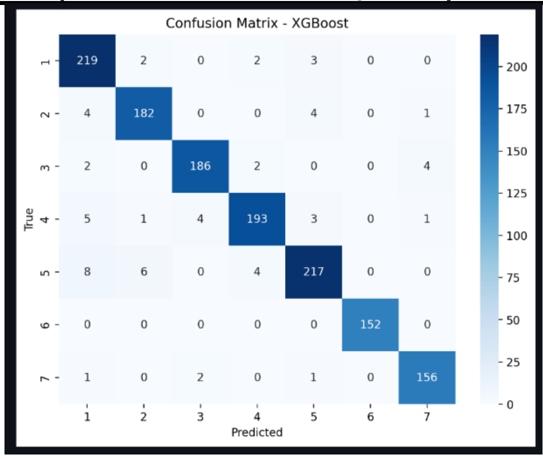


Figure No.7: Confusion Matrix (Decision Tree) **Figure No.8:Confusion Matrix (XGBoost)**

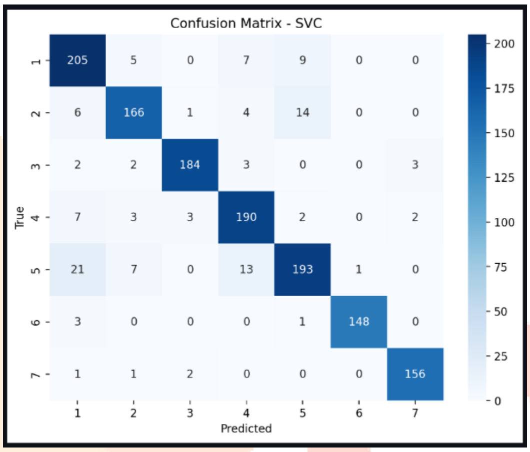


Figure No.9: Confusion Matrix (SVC)

XI. Acknowledgment

We would like to thank our guide **Prof. S. N. Bhadane** for his continuous support and guidance. Also, we are thankful to our project coordinator Prof. S. k. Thakare and we are extremely grateful to **Prof. S. N. Bhadane**, Head of the Department of Information Technology, **Pune Vidyarthi Griha's College of Engineering, Nashik** for his support and invaluable time.

References

[1] Bird, J. J., Kobylarz, J., Faria, D. R., Ekart, A., & Ribeiro, E. P. (2020). Cross-Domain MLP and CNN Transfer Learning for Biological Signal Processing: EEG and EMG. In IEEE Access (Vol. 8, pp. 54789–54801). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/access.2020.2979074

[2] Gaso, M. S., Cankurt, S., & Subasi, A. (2021). Electromyography Signal Classification Using Deep Learning. In 2021 16th International Conference on Electronics Computer and Computation (ICECCO). 2021 16th International Conference on Electronics Computer and Computation (ICECCO). IEEE. https://doi.org/10.1109/icecco53203.2021.9663803

[3] Rakshna, K. B., Tamil Selvan, P., Varshini, S., & Chitra, J. (2023). Pre- Stroke Detection using K- Nearest Neighbour and Random Forest Algorithm. In 2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC). 2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC). IEEE. https://doi.org/10.1109/icaaic56838.2023.10140476

[4] Garg, P., Kumar, P., Shakya, K., Khurana, D., & Roy Chowdhury, S. (2019). Detection of Brain Stroke using Electroencephalography (EEG). In 2019 13th International Conference on Sensing Technology (ICST). 2019 13th International Conference on Sensing Technology (ICST). IEEE. https://doi.org/10.1109/icst46873.2019.9047678