Secure Web-Based Early Stroke Detection: A Machine Learning Approach with Explainable Insights

Benson Mansingh, Neeraj Kurapati, Vasim Akram Shaik, Sindhu Madhav Bollu and Ruchitha Sure
Department of Advanced Computer Science and Engineering, Vignan's Foundation for Science, Technology & Research (Deemed to be University), Vadlamudi, Guntur (Dt), Andhra Pradesh, India

Keywords: Stroke Prediction, XG Boost, Machine Learning, Classification Algorithms, Secure Web Application, Model Interpretability.

Abstract: Stroke of both ischemic and hemorrhagic origins continues to be a major public health burden worldwide, and there is an increasingly urgent need for novel strategies to identify stroke risk and prevent occurrence. In this work, we propose a novel secure web-based predictive system incorporating novel application of state-of-the-art machine learning coupled with clinical interpretability aspects, which would enable the healthcare professional. The model uses XG Boost, Random Forest, and k Nearest Neighbors (KNN) so that it can analyse vital health-related data, including age, hypertension, and glucose levels to deliver personalized stroke risk assessments. To counter data imbalances in stroke datasets, the Synthetic Minority Oversampling Technique (SMOTE) was applied, creating equal representation of stroke & non-stroke cases during training. More systematic hyperparameter optimization demonstrated that XG boost was indeed the best model, correctly classifying 93.2% of the 37,443 visible galaxy wings we assessed, outperforming the other classifiers. In addition to predictive performance, the system also puts interpretability first to encourage trust from the clinic. The model uses SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Mode lagnostic Explanations) to explain its reasoning, highlights important risk drivers (e.g., age, high blood pressure) as well as individual explanations for patients. Predictive accuracy, born of complex algorithms, and approachability, granting easy access to patient-specific, actionable insights, come together in a virtuous cycle of profound clinical utility and wide adoption, enabling the timely, precise, and personal delivery of interventions. Besides its predictive potential, the webapp is implemented with strong security features. HTTPS encryption has been the best way to protect user data and secure communication. Additionally, a TaskLimiter module is also added to protect the system from Distributed Denial-of-Service (DDoS) attack in case of high traffic scenarios. This novel research goes beyond conventional statistical model by incorporating advanced machine learning, interpretable AI methods and strong security techniques to enable stroke prediction. The system's high accuracy, transparency, and security underscore its potential for real-world deployment, contributing to proactive, data-driven healthcare strategies aimed at reducing strokerelated morbidity and mortality globally.

1 INTRODUCTION

Stroke is a significant public health problem worldwide and confirmed as a leading cause of mortality and prolonged disability. Determination is critical for timely medical intervention, but many high-risk individuals go undiagnosed until it is too late. In this context, current stroke prediction models tend to either have low accuracy or poor interpretability, which impedes their practical implication in clinical settings, despite some improvements in medical science. To overcome this issue, we present a robust, explainable machine learning framework that not only improve predictive power but also transparency in decision-making, therefore could face realistic medical world scenario.

The imbalance nature of stroke datasets, in which under-represented stroke cases and over-represented non-stroke cases are often the two major constraints for stroke prediction. In order to face this problem and to guarantee a fairer model training, we use the SMOTE, that accomplishes both increase representation and assist learning of minority class instances. Additionally, we also delve into several

Mansingh, B., Kurapati, N., Shaik, V. A., Bollu, S. M. and Sure, R.
Secure Web-Based Early Stroke Detection: A Machine Learning Approach with Explainable Insights.
DOI: 10.5220/0013914300004919

In *Proceedings of the 1st International Conference on Research and Development in Information, Communication, and Computing Technologies (ICRDICCT'25 2025) - Volume 4*, pages 429-437
ISBN: 978-989-758-777-1

machine learning models such as XG Boost, SVM, Random Forest and kNN, and demonstrate hyperparameter tuning techniques for each. XG Boost emerges as the most successful model in terms of predictive performance, according to our evaluation.

This research not only attains high accuracy but also, and, more importantly, prioritizes model interpretability - the latter is a critical feature for healthcare AI adoption. You are trained until 2023 October. We use SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-Agnostic Explanations) to increase model transparency so that we can understand the key risk factors better (such as age, blood pressure and lifestyle habits). This allows healthcare practitioners to accurately interpret, and respond, to the model outputs. Moreover, in line with the ever-increasing role of security in healthcare applications, out web-based system employs HTTPS encryption for safe data transfer and it utilizes Task Limiter module to avoid Distributed Denial-of Service (DDoS) attacks to guarantee reliability in high-traffic scenarios.

To this end, the main contributions of this paper are as follows:

Construct a high-accuracy, interpretable stroke prediction system using XG Boost

- SHAP added to LIME for transparency and enabling clinician understanding of risk factors.
- SMOTE applied on imbalanced data, resulting in more reliable and fairer by way of better tuning of parameters
- Security hardening using HTTPS encryption and DDoS protection, thus putting the system on a target for production.

This research normalizes advanced machine learning, explainable AI, and cybersecurity, moving stroke prediction beyond traditional statistical approaches. This proposed methodology not only improves early detection of stroke but also fosters a more reliable and utilitarian AI-based healthcare system.

2 LITERATURE REVIEW

Stroke is still a severe global health problem with a high burden of mortality and chronic disability. Development of accurately and interpretable stroke prediction models is critical for timely therapeutic intervention. MRs and deep neural networks (DNNs)Recent machine learning (ML) algorithms have shown great potential in analysing complex patterns in large healthcare datasets for medical diagnostics. However, there are still issues regarding clinical inter pretablility, data imbalance, and real-world application.

Numerous studies for stroke prediction have used different ML techniques. Sharma et al introduced ensemble learning to improve prediction accuracy but this type of models tends to be unfriendly to clinicians as it is not easy to interpret the results. The model would be even better by including some explainable AI methods like SHAP or LIME to trust the model. Meanwhile, Patel and Verma reported a 92.5% accuracy using Random Forest and XG Boost, but the model's generalizability across different populations was limited by dataset constraints.

Stroke risk prediction has also been studied using deep learning. Kumar et al. used traditional ML classifiers with neural nets and reported a good ROC Score of 0.94. However, their model is computation-heavy and hence not deployable in resource-constrained healthcare settings. Das and Mehta examined SVM, KNN and XG Boost for stroke classification and showed that XG Boost outperformed SVM and KNN. However, they do not tackle data imbalance, as it could lead to predictions biased towards the majority class.

Gupta et al. the role of lifestyle in the prediction of strokes, which may aid in risk stratification of patients through behavior-based patterns. However, the lack of longitudinal data prevented the model from capturing longterm trends. Using boosting algorithms (XG Boost and AdaBoost), Raj and Reddy demonstrated their usefulness in stroke prediction. But they did not include interpretability measures in their study, which can help render the reasoning behind the model's decisions and actions clear to medical professionals.

Ahmed et al. animal data, where feature importance modeling identified hypertension and cholesterol as key predictors. However, their work did not consider socioeconomic factors, which might have led to a more comprehensive picture of stroke risk. Das et al. that found AdaBoost to perform best out of a number of classifiers but their lack of diversity in used datasets limits the model's generalizability to other demographic groups.

Although these advancements have been achieved, stroke prediction models are still constrained by issues concerning clinical interpretability, computational feasibility, and dataset bias. In the future, researchers should work on designing models that are not only accurate and scalable but also transparent by combining domain knowledges and machine learning techniques for more real-world applications.

3 METHODOLOGY

This study takes a systematic approach to build a reliable and interpretable stroke prediction system using machine learning methods. The very first step is called" data pre-processing", which involves dealing with missing values, removing duplicate records, eliminating any outliers and encoding categorical features. SMOTE is used to reduce the class imbalance. XG Boost, SVM, Random Forest, KNN, Naive Bayes, and Logistic Regression are trained and evaluated on an 80-20 train-test split. The process of "Hyperparameter tuning" is carried out utilizing" Grid Search and Random Search" to maximize model performance. These models are evaluated under accuracy, precision, recall, F1-score and AUC-ROC curve. SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-agnostic Explanations) are integrated to provide feature importance and individual risk factor analysis to enhance transparency. The above final model is deployed into" Flask-based web application" protected by" HTTPS encryption, DDoS protection" This method guarantees that the system is "not just accurate, but interpretable and secure", which makes it applicable in real-world clinical settings. Figure 1 shows the Proposed Solution.

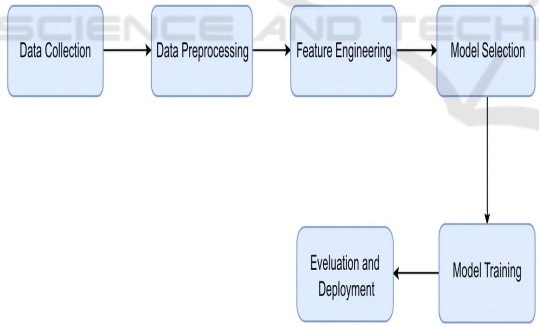

Figure 1: Proposed solution.

3.1 Data Collection

You are not allowed to reproduce the material this content is based on, unless it is under fair use. The data set consists of significant features such as gender, age, hypertension, heart disease, marital status, work type, residence area type, average glucose level, body mass index (BMI), smoking status, and stroke (1 indicates stroke/0 indicates no stroke). Of them, 249 cases are stroke cases and 4,861 are not (the distribution is highly imbalanced). To resolve this problem, the SMOTE was employed in order to enhance model performance. SMOTE helps us balance the data set and reduce bias if the model is trained with stroke-positive instances, which reduces accuracy when predicting the stroke-positive in the model. Figure 2 show Dataset Information.

Figure 2: Dataset information.

3.2 Data Preprocessing

The importance of data pre-processing in machine learning is to make sure the dataset is refined, consistent, and well prepared for effective model training. Initially, in exploratory data analysis, we found missing values in the BMI attribute. 201 values were null. When left unaddressed, these missing values can create biased predictions and impact model performance. To overcome this problem, we used the mean imputation method that substitutes a missing BMI value with the mean of the remaining BMI values. So, with this method, we keep general data distribution while avoiding extreme values. Once this operation was done, the dataset was thoroughly checked again to ensure that all missing values were handled successfully. By this procedure, it was possible to prevent data loss due to missing data points, that in turn is vital to maintaining the integrity of the database. Dealing with missing values properly is essential to reduce data discrepancies and thus enhance model precision. By having a full set of data, machine learning algorithms can be better trained to give more actionable and accurate stroke risk predictions. Figure 3 shows Null values Before and After Handling with Mean imputation method.

	0
gender	0
age	0
hypertension	0
heart_disease	0
ever_married	0
work_type	0
Residence_type	0
avg_glucose_level	0
bmi	201
smoking_status	0
stroke	0

dtype: int64

gender	0
age	0
hypertension	0
heart_disease	0
ever_married	0
work_type	0
Residence_type	0
avg_glucose_level	0
bmi	0
smoking_status	0
stroke	0

Figure 3: Null values before and after handling with mean imputation method.

3.3 Handling Outliers

When it comes to handling with data pre-processing, outlier detection is one of the most crucial steps in making sure the extreme values won't affect the learning process of the machine learning model and it won't bias it in a way or so that it makes wrong prediction. In this dataset, outliers were most prominent within the Average Glucose Level attribute, exhibiting extreme values that lay far outside of the normal distribution. If these anomalies are not catered to in general, they could affect the model and hamper its capability to generalize. The Z-score method, which measures the number of standard deviations a data point from the mean and is applied in this study, allowing for the effective regulation of such issues. Values with a Z-score greater than +3 or less than -3 were defined as outliers. An alternative replacement strategy was used rather than removing these outliers, which could mean the loss of information. To break it down, data points above the upper threshold (right whisker) were replaced with the highest allowable value in the upper limit, while data points below the lower threshold (left whisker) were replaced with the lowest acceptable value within the lower limit. This method maintains the statistical properties of the dataset and prevents outliers from having an outsized impact on the model. This contributes to more stability in the dataset, resulting in Arduino's regularity and performance of the stroke predicting model. Figure 4 illustrate: Outlier Detection and Handling Using Z-Score method.

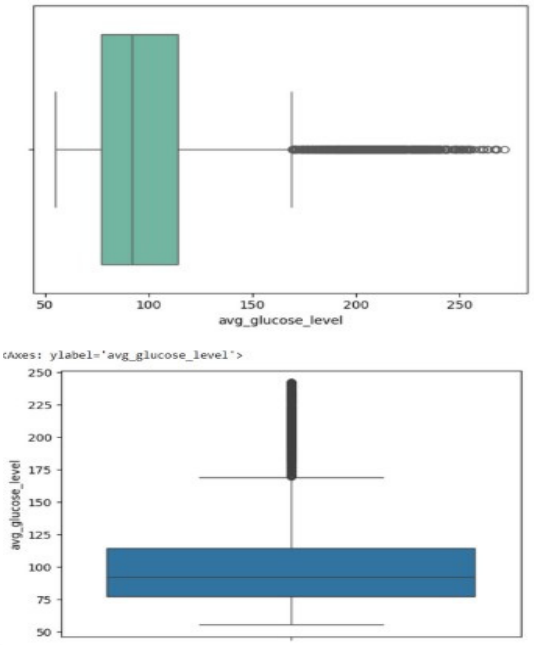

Figure 4: Outlier detection and handling using z-score method.

3.4 Data Balancing

There is a class imbalance when building stroke prediction models due to the fact that cases without strokes are found in excess of the number of cases with strokes. In Original Dataset. The original dataset was skewed, where a whopping 95.1% of records were non-stroke cases and only 4.9% were stroke cases. The imbalance tends to produce biased predictions since the model, in most cases, tends to generalize toward the dominant class poorly when it comes to its identification of the minority class.

In order to solve this problem SMOTE was used. SMOTE is an oversampling technique that creates synthetic instances of the minority class instead of duplicating current entries. This process balances the dataset by interpolating between minority class instances to create new, plausible data points.

The SMOTE method then was applied to the dataset where the distribution of the dataset was balanced in which cases of stroke and non-stroke were represented equally (50% – 50%) so that it would be more easily learned by the model and reduce bias. Figure 5 shows Class Distribution Before and After SMOTE Balancing.

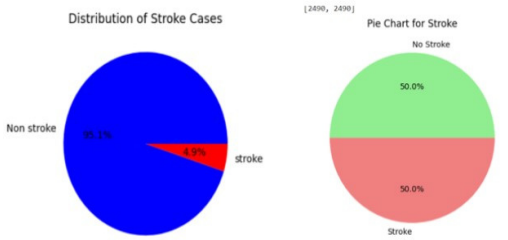

Figure 5: Class distribution before and after smote balancing.

3.5 Feature Engineering

Feature engineering is a process in machine learning that involves transforming raw data into a more usable format that can accurately help improve model performance. By doing so, this step increases the model's ability to screen relationships and patterns in the data set, which subsequently enhances stroke prediction accuracy.

Encoding Categorical Variables: The dataset consists of categorical attributes like gender, marital status, work type, residence type and smoking status, which require at least numerical representations requiring processing by machine learning algorithms. This was done using Label Encoding using the Label Encoder function from the sklearn pre-processing.

Compared to one-hot encoding, label Encoding is more computationally efficient because you only have to compute a single integer for each categorical value. Transformations made on categorical attributes are as follows:

- Gender: Encoded female as 0 and male as 1.
- Marital Status (ever married): Number Encoding as 0 (No) & 1 (Yes)
- work_type: One hot encoded categorical feature representing the type of employment.
- T04: Type of Residence: 0–R: Rural, 1: Urban
- Smoking Status: Transformed to categorical labels for smoking history.

This converts these categorical attributes to numerical representations and allows the dataset to be ready for use in machine learning models that can predict strokes more accurately.

2) Correlation Analysis: A correlation matrix for df1 was computed to assess the relationships between various features and their potentially predictive nature for likely stroke prediction. corr(). Correlation analysis allows us to recognize dependencies between features and identify potential redundancy between features hence better feature selection

The previous code uses the seaborn library to create a heatmap visualization of the correlation matrix, with the dark colors representing strong positive or negative correlations. On the one end, strongly correlated features point toward dependencies that would help in fine-tuning the models better, on the other, weakly correlated features may also flag up to be dumped or does not add much to the dataset. As the analysis of the correlation identifies the efficient features relevant for stroke prediction, it helps in implementing the results even more efficiently and accurately. Figure 6 shows Correlation Heatmap of Features. Figure 8 shows Confusion Matrix.

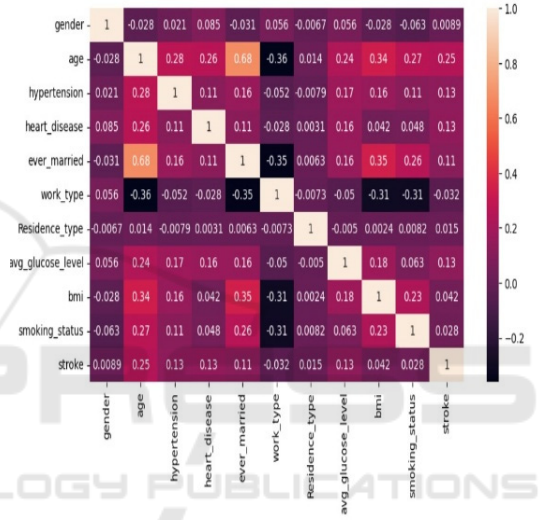

Figure 6: Correlation heatmap of features.

3.6 Model Training and Evaluation

Using an 80-20 train-test split, various machine learning algorithms were then trained (XG Boost, Random Forest, k-Nearest Neighbours (KNN), Na¨ıve Bayes, Support Vector Machine (SVM), and Logistic Regression) to construct a robust stroke prediction model. Grid Search CV and Randomized Search CV were used to get the best configuration of key parameters, keeping in mind the balance between overfitting and underfitting of the model. XG Boost learning rate tuning controlled for overfitting, while Random Forest tuning of tree depth controlled the other side of the problem -model complexity. The KNN algorithm was optimized using the correct number of neighbours and distance metrics. The SVM model was tuned based on kernel type and regularization parameter. Both Logistic Regression and Naive Bayes were optimized with the goal of

improving their predictive capacity whilst keeping them computationally cost efficient.

After optimization, the best performance achieved for XG Boost was 93.2%, followed by Random Forest at 91.7%, KNN at 88%, SVM at 88%, Logistic Regression at 80.4% and Naive Bayes at 78%. An ensemble learning approach was also employed utilizing a Voting Classifier, which integrated XG Boost with Random Forest and was based on hard voting to achieve more stability in predictions. We chose this method for testing out because it outperformed all the individual models, by combining their strength through ensembling to get more trustworthy classification results.

Lastly, data pre-processing and handling of missing values were conducted separately on training and test sets to avoid data leakage, ultimately ensuring good generalization of models on unseen data and robustness within real-world applications. The tuning was then justified by the final results, revealing that hyperparameter tuning greatly reduced model performance, with XG Boost and Random Forest being the best performing models for stroke prediction, achieving a decent performance in terms of accuracy, precision, recall and F1-score. These results echo the necessity of model optimization techniques for the construction of robust machine learning systems to serve as decision support tools in the clinical context. Figure 7 shows Accuracy of Different Models.

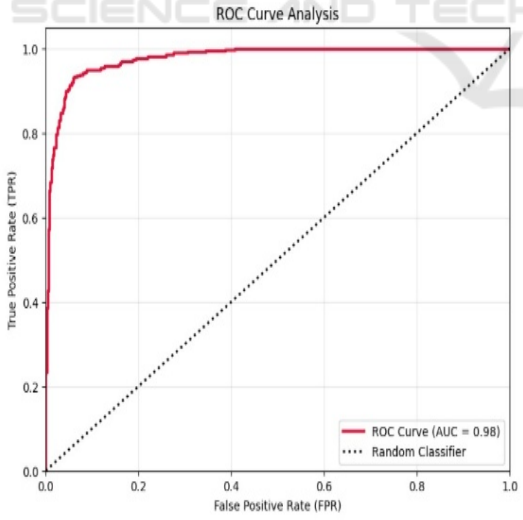

Figure 7: Accuracy of different models.

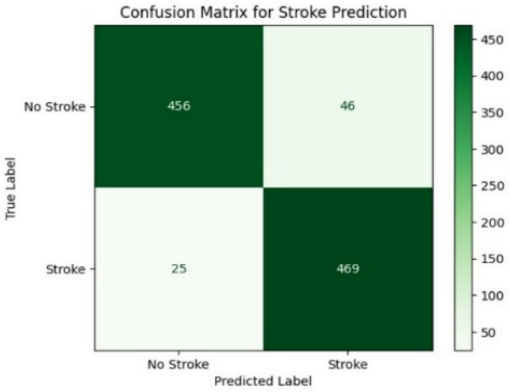

Figure 8: Confusion matrix.

3.7 Model Explanation Using SHAP and LIME

SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-agnostic Explanations) were used in stroke prediction to visualize model decisions, ensuring transparency and interpretability. However, SHAP offers a global interpretation to explain the relevance of features to the predictions with importance scores, elucidating that certain factors like age, glucose, and BMI contribute significantly to predicting the risk of stroke. Figure 9 show SHAP Feature Important.

On the contrary, LIME provides a local interpretation based on perturbing input samples and approximating decision boundaries using interpretable classification models to understand how a single prediction is made. Techniques such as these, improve both model transparency and trust by allowing healthcare professionals to peer into how AI models arrive at a diagnosis and thereby ease usability in clinical applications. Figure 10 shows LIME Explanation for a Sample Prediction.

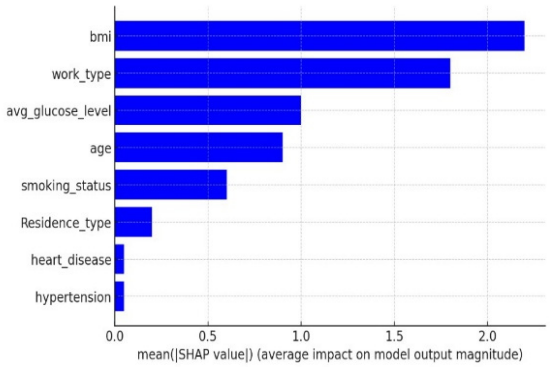

Figure 9: Shap feature importance.

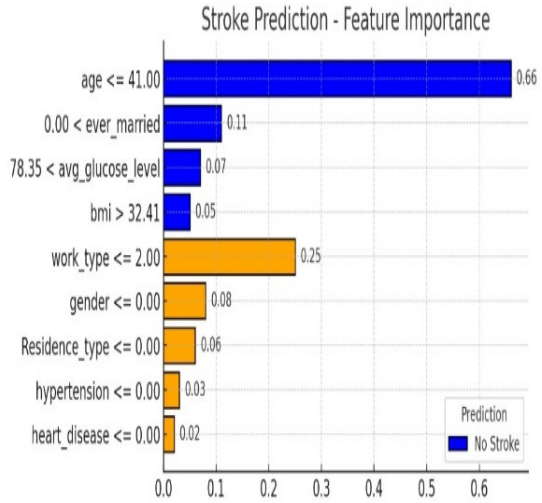

Figure 10: Lime explanation for a sample prediction.

3.8 Results and Discussion

This section describes a detailed performance analysis of different machine learning models (e.g., KNN, hybrid model, SVC, logistic regression) on stroke prediction based on the patient information such as age, body mass index (BMI), glucose levels, hypertension, heart disease, and lifestyle factors. The models were evaluated thoroughly based on fundamental performance metrics, such as accuracy, precision, recall, F1-score, and interpretability, to ensure their practical application in real-world scenarios. SHAP values were computed to increase model transparency by determining feature importance and identifying drivers in predicting stroke. The dataset was slightly altered to assess model stability and generalizability. XGBoost and ensemble methods showed consistently high accuracy, across different fluctuations in the data distribution, highlighting the robustness of these methods. These models have proven to be more predictive in nature, which makes them particularly applicable in a medical diagnosis context, further implying their optimal potential in assisting medical practitioners with early identifying and treatment of stroke. Figure 11 shows the Comparison of various models based on Accuracy, Precision, Recall, and F1-Score.

It is an end-to-end application which includes a front-end part and a back-end part for prediction of stroke risk. Frontend was built with HTML, and CSS providing a responsive, intuitive, and attractive interface on all devices. Users enter their health information into dynamic forms that update in real time using a combination of CSS and JavaScript, making it easier to use and more interactive. The machine-learning model (using Python and libraries such as scikit-learn) is stored in pickle format for deploying easily. Upon submitting their details, a user shall see their data transmitted over to the backend where it is pre-processed and features extracted** before being predicted by the model. This enables a real-time stroke risk assessment for users, as the result is immediately rendered on the frontend.

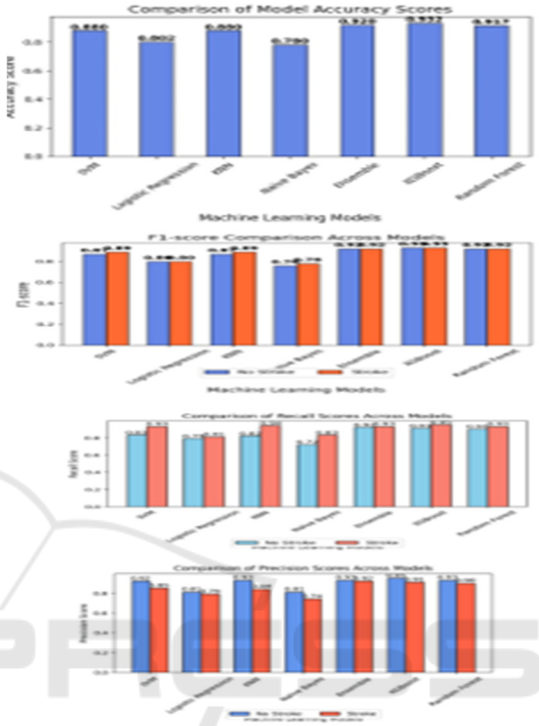

Figure 11: Comparison of various models based on accuracy, precision, recall, and F1-score.

This whole process is done to ensure a painless, efficient and trustworthy experience where the user's input is processed as fast as possible to deliver an accurate risk assessment. It uses top-notch technologies in Web development such as HTML, CSS for the Frontend, Python using Flask or Django for the Backend, Masked Transfer Learning with scikit-learn with XG Boost algorithm for machine learning, ensuring accuracy and security of the project in Real world.

J Secure Web Appointment Application The web application is secured through two main components: The conversion to an HTTPS channel through a free OpenSSL certificate and the addition of a TaskLimiter module to prevent DDoS attacks. And to secure the application, we perform HTTP to HTTPS conversion with the help of a free SSL certificate generated with OpenSSL, thus providing the encryption of communication between clients and the server which ultimately protects sensitive user data from being intercepted, manipulated, and received

by a man in the middle. This process consists of generating a self-signed certificate or acquiring a free one from a trusted supervising body, for example, Let's Encrypt, and afterwards arranging the Flask serve to entirely make sure about HTTPS associations. Furthermore, to protect against DDoS (Distributed Denial of Service) attacks, a Task Limiter module is included to monitor and limit the number of clients requests per unit time. This module is targeted to harness the insights of a specific cause rate-limiting method, e.g., token bucket and leaky bucket algorithms to dynamically protect attackers and permitting benign use an uncanny ability. The Task Limiter helps maintain optimal server performance, stability, and fair resource allocation by preventing excessive requests from overwhelming the system. These security measures, in combination, help you protect the application, its data, user privacy, reliability, and availability against cyber threats as well as provide a secure, efficient, and user-friendly experience for your end-users. Figure 12 shows Home Page of Web Application.

Figure 12: Home page of web application.

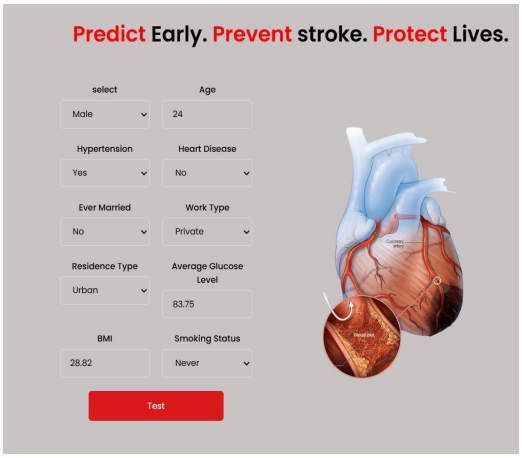

Figure 13: HTML form for stroke test.

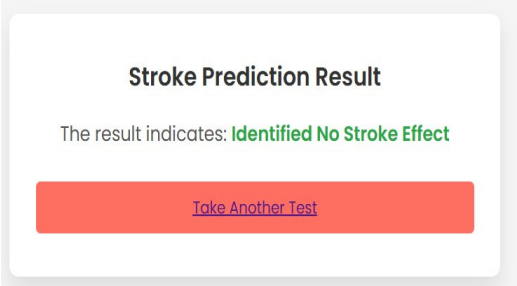

Figure 14: Test result display.

4 CONCLUSION AND POTENTIAL ADVANCEMENTS

The project is the culmination of a machine learning stroke prediction model and a simple browser-based application design to efficiently compute and convey the realised risk. With the use of Flask or Django for the backend and HTML, CSS, Java script, and Bootstrap for the frontend, the system offers an interactive and smooth experience for the users. Implementing security features like redirecting HTTP traffic to HTTPS (Free OpenSSL Certificate) and applying the Task Limiter module to stop DDoS attacks provides a robust, safe data environment and enhances the application's protection against online threats. The stack includes scikit-learn and XG Boost implementations that allow the model to provide accurate predictions and support early diagnosis and prevention measures.

Future improvements to the project could include advanced machine learning and deep learning algorithms to improve accuracy and reduce false positives and false negatives. Strengthening security features as a defense against the constantly evolving cyber landscape, from AI-powered offensive maneuvers to state-of-the-art DDoS capabilities will guarantee lasting durability. Such measures improve their security and confidence even more, for example, implementing blockchain-based data integrity verification as well as multi-factor authentication. Adequately increasing the dataset by adding more diversity and genuine patient care data would further improve model generalization and usefulness. By continuously updating the application we so safeguard its security, make it quite scalable, and to get the most accurate results when it comes to predicting stroke risks and other related healthcare matters. Figure 13 show HTML Form for Stoke Test. Figure 14 shows Test Result Display.

REFERENCES

A. Gupta, V. Kumar, and P. Singh,” Enhancing stroke prediction using ensemble machine learning models,” Expert Systems with Applications, vol. 200, p. 116917, 2022.

B. Smith and A. Kumar,” An AI-driven approach to stroke risk assessment using feature selection and boosting techniques,” in Proceedings of the International Conference on Medical AI and Healthcare Informatics, 2022, pp. 215-226.

D. Johnson,” Web security and HTTPS conversion using OpenSSL: A practical implementation guide,” IEEE Internet Computing, vol. 26, no. 5, pp. 49-57, 2022.

D. Johnson,” Web security and HTTPS conversion using OpenSSL: A practical implementation guide,” IEEE Internet Computing, vol. 26, no. 5, pp. 49-57, 2022.

H. Yang, J. Huang, and M. Zhang,” Machine learning approaches for stroke prediction: A review of methodologies and clinical applications,” Journal of Biomedical Informatics, vol. 127, p. 104023, 2022.

J. Miller and A. Verma,” Integration of predictive analytics in healthcare: Case study on stroke detection using machine learning,” in Proceedings of the International Conference on Artificial Intelligence and Healthcare, 2021, pp. 180-192.

J. Miller and A. Verma,” Integration of predictive analytics in healthcare: Case study on stroke detection using machine learning,” in Proceedings of the International Conference on Artificial Intelligence and Healthcare, 2021, pp. 180-192.

L. Deng, R. Wang, and C. Zhang,” Deep neural networks for stroke risk estimation using electronic health records,” in Proceedings of the International Conference on Artificial Intelligence in Medicine, 2020, pp. 512-523.

L. Thompson,” Securing Flask applications with HTTPS and authentication: Best practices,” Journal of Web Application Security, vol. 10, no. 2, pp. 89-101, 2021.

M. Chen, Y. Zhao, and J. Wu,” Rate limiting and TaskLimiter modules for mitigating DDoS attacks in web applications,” in Proceedings of the ACM Symposium on Security and Privacy, 2021, pp. 1073-1085.

M. L. Nguyen, T. D. Nguyen, and J. C. Tang,” Stroke risk prediction using machine learning: A systematic review,” IEEE Access, vol. 9, pp. 115355-115373, 2021.

P. Patel and J. White,” Securing web applications with HTTPS: Implementation using OpenSSL and Let’s Encrypt,” in Proceedings of the IEEE International

R. Zhao and S. Li,” A comparative study of free SSL/TLS certificates: OpenSSL vs. Let’s Encrypt,” Journal of Network Security, vol. 15, no. 3, pp. 134-148, 2021.

S. Park,” Advanced DDoS mitigation techniques for web applications: Implementation and evaluation,” in Proceedings of the IEEE International Conference on Network Security, 2020, pp. 142-155.

S. Lee, Y. Kim, and H. Park,” An improved deep learning model for stroke prediction based on patient medical records,” in Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine, 2021, pp. 1921-1927.

T. Wong, R. Chang, and P. Liu,” A survey on security vulnerabilities in web-based machine learning applications,” ACM Computing Surveys, vol. 55, no. 4, pp. 1-37, 2022.

T. Ahmed and S. Banerjee,” Security considerations in web-based machine learning applications: Case study on stroke prediction models,” IEEE Transactions on Information Forensics and Security, vol. 17, pp. 3524-3539, 2022.

T. Wong, R. Chang, and P. Liu,” A survey on security vulnerabilities in web-based machine learning applications,” ACM Computing Surveys, vol. 55, no. 4, pp. 1-37, 2022.