TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

KATHMANDU ENGINEERING COLLEGE

DEPARTMENT OF COMPUTER ENGINEERING

MINOR PROJECT REPORT

ON

STROKEOPUS: BRAIN STROKE PREDICTION USING MACHINE LEARNING

[Code No: CT654]

By:

Shreya Maharjan	–	KAT077BCT079
Simran Maharjan	–	KAT077BCT083
Sneha Dhital	–	KAT077BCT084
Sornika Koirala	–	KAT077BCT088

Kathmandu, Nepal

March ,2024

TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING

KATHMANDU ENGINEERING COLLEGE

DEPARTMENT OF COMPUTER ENGINEERING

STROKEOPUS: BRAIN STROKE PREDICTION USING MACHINE LEARNING

[Code No: CT654]

PROJECT REPORT SUBMITTED TO

THE DEPARTMENT OF COMPUTER ENGINEERRING

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR

THE BACHELOR OF ENGINEERING

Submitted by:

Shreya Maharjan	–	KAT077BCT079
Simran Maharjan	–	KAT077BCT083
Sneha Dhital	–	KAT077BCT084
Sornika Koirala	–	KAT077BCT088

TRIBHUVAN UNIVERSITY

Kathmandu Engineering College

Department of Computer Engineering

Kathmandu, Nepal

March, 2024

ACKNOWLEDGEMENT

We would like to extend our cordial gratitude to the Institute of Engineering (IOE) for providing us with the opportunity to develop a project under the academic requirement as a minor project. We are also very grateful to our project supervisor and the **Er. Sudeep Shakya,** Head of Department of Computer Engineering and **Er. Kunjan Amatya,** Deputy Head of Computer Engineering for their scholarly guidelines and suggestions during the development of our project. We are equally indebted to **Er. Anju Khanal** and **Er. Krista Byanju** for providing us with guidelines throughout the project development process which has been a great help to start our project.

Moreover, we would like to thank Department of Computer Engineering, Kathmandu Engineering College for providing us with an opportunity to explore our true potential and build a project of our interest which has highly helped us to implement the knowledge and skills gained over the years as the minor project.

Without their guidance, it would be an uphill task. So finally, we would give our sincere thanks to all those who were involved in any way in our project and are grateful for your contribution.

ABSTRACT

Stroke is the second leading cause of death and disability worldwide. The main objective of this project is to detect the possibility of a stroke as early as possible so as to reduce the risk and severity and suggest some preventive measures of a stroke. Initially, all the data were collected from Kaggle named “Brain stroke prediction dataset”. The dataset was last updated on July 16, 2022, by Izzet Turkalp Akbasli. The dataset contains 10 attributes: age, gender, heart disease, average glucose level, body mass index, ever married, work type, hypertension, residence type and smoking status. The dataset was divided into training and testing data in the ratio of 9:1. Here, machine learning techniques were clubbed to analyze and predict the probability of having a stroke with the help of decision tree (DT) and random forest (RF) algorithms which gave a validation accuracy of 94.8% and 97% respectively. Among the algorithms, since random forest gave a higher accuracy, we chose RF to be implemented in the system. For better understanding, use case diagram and sequence diagram were employed. From this system, user can receive guidance on lifestyle modification such as adopting healthy diet, exercising regularly, managing blood pressure, cholesterol level, quitting smoking and controlling diabetes according to their respective impact factor in the result predicted.

Keywords: *Stroke Prediction, Machine Learning, Decision Tree, Random Forest*

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

aHR	Adjusted Hazard Ratio
ASM	Attribute Selection Measure
BMI	Body Mass Index
CI	Confidence Interval
DALYs	Disability-adjusted life years
DT	Decision Tree
k-NN	k-Nearest Neighbors
RF	Random Forest
SVM	Support Vector Machine
UI	User Interface
WEKA	Waikato Environment for Knowledge Analysis

CHAPTER 1: INTRODUCTION

1.1 BACKGROUND THEORY

A stroke, also known as brain attack, happens when blood flow to the brain is blocked or when a blood vessel in the brain bursts. Stroke is divided into ischemic and hemorrhagic. Ischemic strokes occur when the blood supply to part of the brain is interrupted or reduced. About 87% of all strokes are ischemic. Hemorrhagic strokes occur when a blood vessel that supplies the brain ruptures and bleeds. When an artery bleeds into the brain, brain cells and tissues do not get oxygen and nutrients. About 13% of all strokes are hemorrhagic [1]. To work properly, our brain needs oxygen and nutrients. Preventing brain tissue from getting oxygen and nutrients, brain cells begin to die within minutes, which causes stroke.

Stroke is one of the most serious diseases worldwide, directly or indirectly responsible for a significant number of deaths. It is an important cause of disability among adults and is second leading cause of death worldwide. The most common types of disability after stroke are changes to speech, learning and understanding, and weakness or paralysis on one side of the body.

Early prediction of the stroke helps the patient to take the medical treatment and they can avoid the risk of stroke. Patient identified as high-risk can receive guidance on lifestyle modification such as adopting healthy diet, exercising regularly, managing blood pressure, cholesterol level, quitting smoking and controlling diabetes. These interventions can reduce the stroke occurrence and contribute to long term stroke prevention. So early detection of stroke is very important.

The focus of our work is to develop accurate and reliable model that can analyze various data source to predict the likelihood of a stroke occurrence in an individual. Here, stroke is predicted in terms of many symptoms like age, gender, heart disease, average glucose level, hypertension, ever married, body mass index, smoking status, work type and residence type.

1.2 PROBLEM STATEMENT

As stroke being the second leading cause of death and disability worldwide**,** there is not enough talks and information in the context of Nepal. Stroke contributed 7.6% of total deaths and 3.5% of total DALYs in Nepal, with a higher burden among the male and old age population. Intracerebral hemorrhage was the dominant type of stroke in Nepal with the highest proportion of deaths and DALYs. High systolic blood pressure was contributing the maximum DALYs due to stroke in Nepal. Conclusion is hemorrhagic stroke causes high mortality and DALYs in Nepal. Most of the burden of stroke is attributed to high blood pressure in Nepal [2].

Especially in developing countries like Nepal where manpower is the most important factor for the development and betterment of the country, it indirectly affect the prosperity of country. Only a one to one appointment with the doctors is a possible way to determine the risk of stroke. However, in this process there may be some human errors in the examination done by the doctors. The lack of proper technology related to this problem also provides difficulty to the people. There is not much awareness in this subject among the people of Nepal.

1.3 OBJECTIVES

The main objectives of this project are:

- To detect the possibility of a stroke as early as possible so as to reduce the risk and severity of a stroke.
- To assist in implementing appropriate preventive strategies for high-risk individuals which may include lifestyle modifications and clinical decision support to reduce the risk factors associated with strokes.

1.4 SCOPE AND APPLICATION

1.4.1 Scope

By observing the trends of deaths and Disability-Adjusted Life Years (DALYs) due to stroke from 1990 to 2017 and the contribution of major risk factors to stroke burden in 2017, it was found that stroke contributed 7.6% of total deaths and 3.5% of total DALYs in Nepal, with a higher burden among the male and old age population. Our project aims to use various technologies and algorithms to identify individuals who may be at a risk of experiencing a stroke. Hence, the project has the scope in the field of healthcare and medical research.

1.4.2 Application

The applications of our project is listed below:

- To analyze various risk factors such as age, medical history and lifestyle choices to assess an individual's likelihood of experiencing a stroke.
- To provide early warnings of potential stroke events by continuously monitoring an individual's health parameters, such as blood pressure, heart rate, cholesterol levels, and other relevant data.
- To provide recommendation which may include lifestyle modifications, medication adherence, dietary changes, exercise routines, and stress management strategies.
- To contribute to research efforts and population health studies from the data collected from our report.

CHAPTER 2: LITERATURE REVIEW

Stroke is ranked as the second leading cause of death worldwide with an annual mortality rate of about 5.5 million. Not only does the burden of stroke lie in the high mortality but the high morbidity also results in up to 50% of survivors being chronically disabled. Thus, stroke is a disease of immense public health importance with serious economic and social consequences.

According to the definition proposed by the World Health Organization in 1970, “stroke is rapidly developing clinical signs of focal (or global) disturbance of cerebral function, with symptoms lasting 24 hours or longer, or leading to death, with no apparent cause other than of vascular origin” [3].

Jeena *et al.* (“A Snapshot of the Burden, Epidemiology, and Quality of Life”, 2018) predict the chances of a stroke via Support Vector Machine (SVM) using the biological risk factors. After preprocessing, 12 risk factors were given as input with 350 samples. SVM had been implemented through MATLAB with multiple different kernel functions. The error rate was used to assess the classifier’s efficiency, whereas validation necessitated the calculation of sensitivity, specificity, accuracy, precision, and F1 score. They obtained the best accuracy through Linear SVM Classifier (accuracy of 91%), with the results being evaluated on a spectrum of patients of different age groups [4].

Minhaz *et al.* (“A hybrid machine learning approach to cerebral stroke prediction based on imbalanced medical dataset”,2019) collected data of 5110 healthy and unhealthy subjects and considered various features, including age, hypertension, BMI level, heart disease and smoking status to predict stroke incidence as a binary classification problem. They employed ten classifiers including Logistic Regression, Stochastic Gradient Descent, Decision Tree Classifier etc and finally aggregated the results of the base classifiers by using the weighted voting approach to reach the highest accuracy. Performance measures including accuracy, area under the curve, precision and recall were the highest for weighted voting approach [5].

Shoily *et al.* (“National Institute of Neurological And Stroke Disorder”, 2023) combed through multiple datasets for a sample of size 1058 for different types of strokes (ischemic, hemorrhagic, brain stem and mini-stroke) with a total of 28 features and prepared them to use with the WEKA toolkit. They then used inbuilt algorithms from the tool set like Naive Bayes, Random Forest and J48 (The popular decision tree algorithm C4.5 is implemented in WEKA as a classifier named J48) and k-NN. Finally, they compared the accuracy, precision, recall and F1-score to conclude that Naive Bayes gave the best result with an accuracy of 85.6 [6].

The proposed system acts as a prediction support machine and will prove as an aid for the user with diagnosis. The algorithms used to predict the output have potential in obtaining a much better accuracy then the existing system. The aim of our project is to have high performance and accuracy rate for brain stroke prediction. In our system, data and information collected for prediction is easily available to the users. Our system will provide users with precaution that can be taken to reduce risk factor.

CHAPTER 3: METHODOLOGY

3.1 PROCESS MODEL

In order to attain or develop a system having great quality and efficiency it is important to have a process model in place as it helps to transform the idea of a project into a functional and completely operational structure covering the technical aspects of system development along with process development, change management, user experience, and policies.

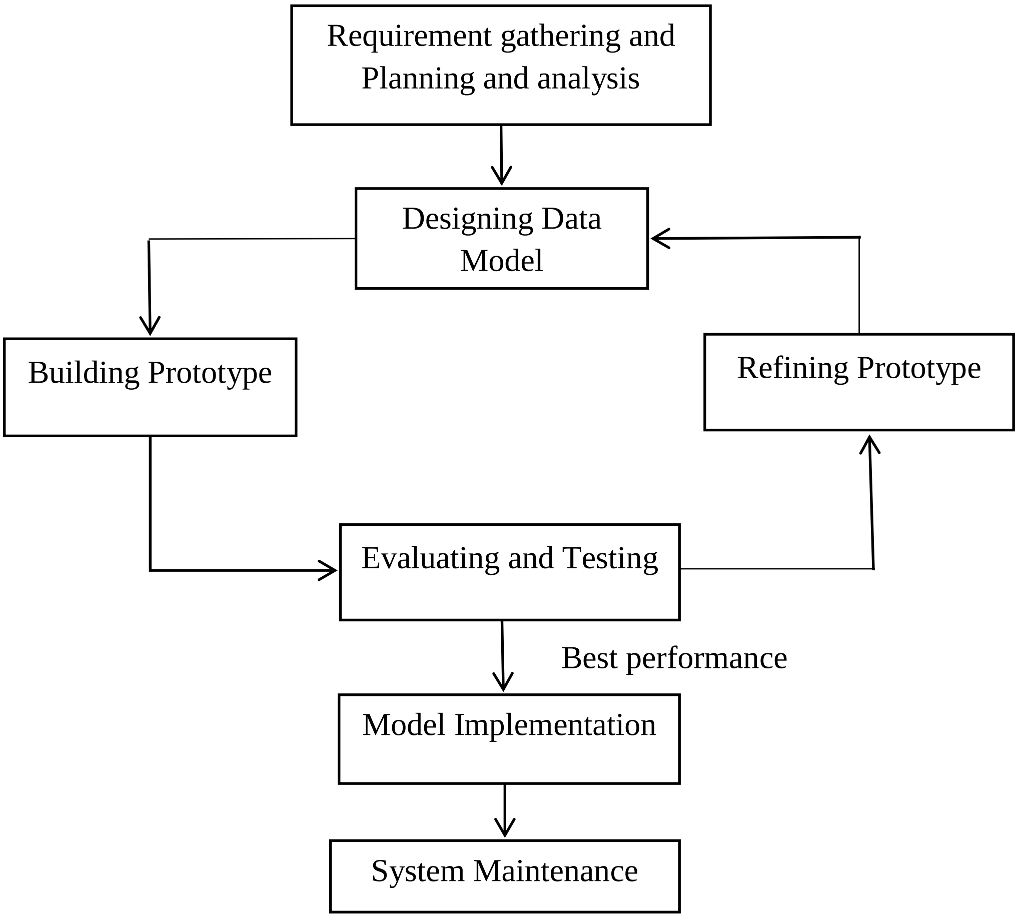


Figure 3.1 Prototyping process model

Considering this we have chosen Prototyping Model for our system development where a prototype is built, tested, and reworked until an acceptable prototype is achieved. It is an iterative, trial and error method where the system is redefined until it meets the desired requirements repeatedly and a final acceptable prototype is achieved which forms the basis for developing the final product.

Prototyping Model is a software development model in which prototype is built, tested, and reworked until an acceptable prototype is achieved. It also creates base to produce the final system or software. It works best in scenario where the project's requirement are not known in detail. It is an iterative, trial and error method which takes place between developer and client.

The various development phases that are to be fulfilled while working on a machine learning problem considering this model are:

Step 1: Requirement gathering, planning and analysis:

Based on this machine learning problem all data and information that are needed for the calculation, also called datasets, are collected and required. Identifying and monitoring the system's technical and economic aspects as well as the timing complexities are observed in this phase. Similarly, domain analysis and scenario planning are also considered.

Step 2: Quick design of data model:

All the data acquired in the first stage are in raw form i.e., data are unmanaged and some data are unwanted. So, they need to be converted into suitable form before manipulating which involves various steps. By analyzing the raw data, we clean the data by removing duplicate data, and irrelevant observations, handling missing data and validating the data. Finally, a valid dataset in order to perform the required data analysis for the machine learning problem.

Step 3: Building an initial prototype:

Here, the valid dataset that is obtained after data cleaning is processed. A model is developed using the required algorithm to process the data and a result is obtained. The user interface (UI) is also developed and designed to ensure usability and a seamless user experience. Interactive elements are created for better user experience. The result of this initially built prototype might not be the desired one so refining need to be done.

Step 4: Reviewing and refining prototype:

According to the results obtained from the given prototype which are tested and reviewed, we again refine this prototype. This process is repeated until the model produces satisfiable accuracy. For this we shall employ feature engineering, feature selection, optimizing the hyper parameters of the given model, cross validation technique etc. The model is refined and necessary changes are made until it is good enough to be ready for the further production and development. The testing data is used to evaluate the accuracy of the model. Similarly, for the front end part, designing issues, usability problems or areas for improvement are identified and are worked upon. If the user is not happy with the current prototype, you need to refine the prototype with the user’s feedback and suggestions. This phase will not be over until all the requirements specified by the user are met. Once the user is satisfied with the developed prototype, a final system is developed based on the approved final prototype.

Step 5: Implementing model and maintenance:

Once the final system is developed based on the final prototype, it is thoroughly tested and deployed to production which is able to make predictions for new data in a required level of accuracy. Similarly, the system components for data collection, storage, and accessibility can also be developed for further processing and improving the model continuously based on the collected data.

3.2 SYSTEM BLOCK DIAGRAM

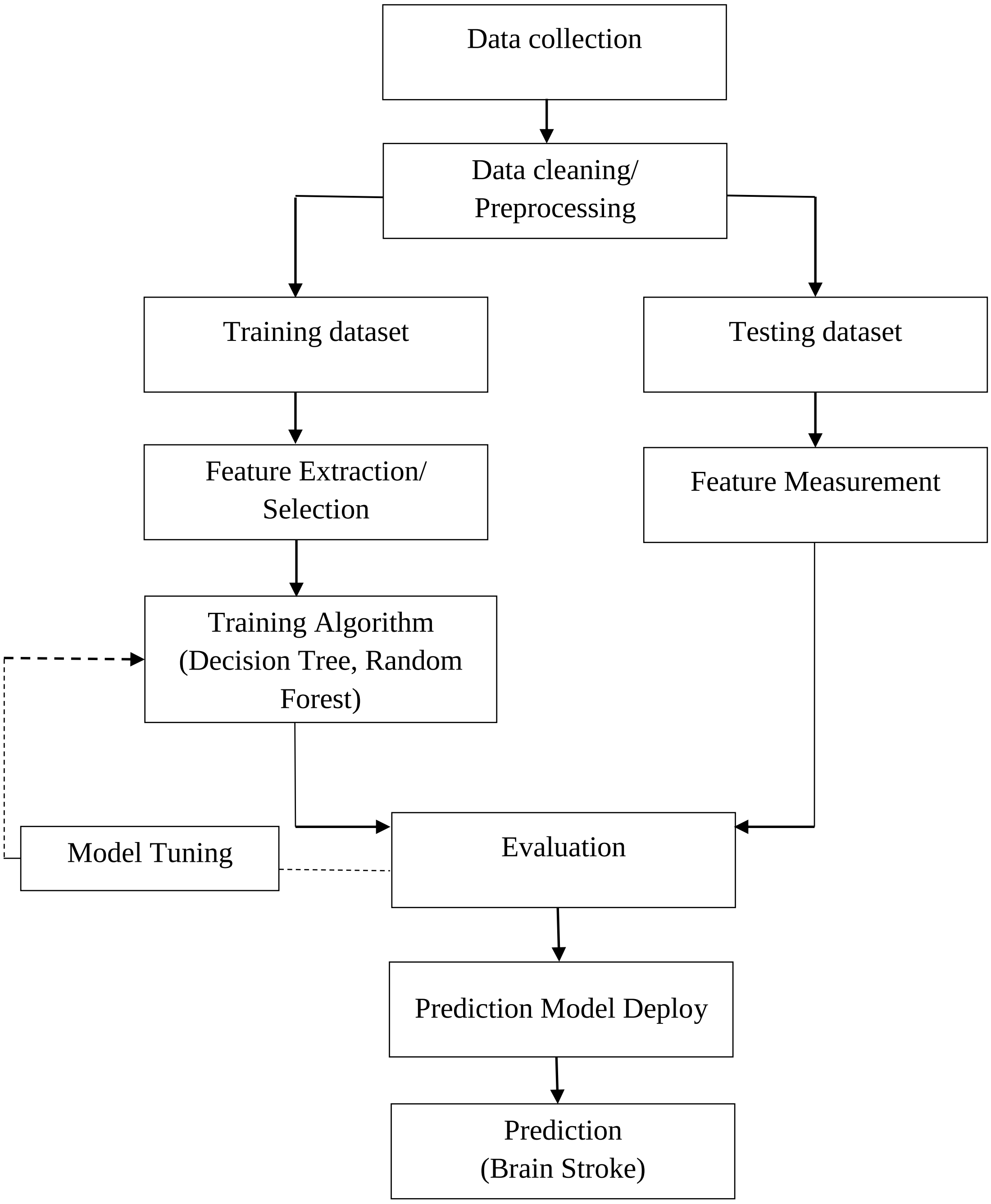

Data collection
Data cleaning/ Preprocessing
Training dataset
Testing dataset
Feature Extraction/ Selection
Feature Measurement
Training Algorithm (Decision Tree, Random Forest)
Model Tuning
Evaluation
Prediction Model Deploy
Prediction (Brain Stroke)

Figure 3.2 Block diagram showing the steps for brain stroke prediction

Initially, all the data are collected from Kaggle named “Brain stroke prediction dataset”. The dataset was last updated on July 16, 2022, by Izzet Turkalp Akbasli. The dataset contains 10 attributes: age, gender, heart disease, average glucose level, body mass index, ever married, work type, hypertension, residence type and smoking status. These data are in raw form i.e., they are umanaged and unwanted. They need to be converted in suitable form before manipulating which involves various steps. By analyzing the raw data, we clean the data by removing duplicate data and irrelevant observations, handling missing data and validating the data. Finally, a vaild dataset in order to perform the requires data analysis for the machine learning problem is created.

Then the obtained dataset is split into training and testing dataset. The training dataset is pre-processed and relevant features are selected. Then the data is fed into the algorithm. Considering our problem, we have used decision tree classifier and random forest algorithms for training the dataset which is further sent for evaluation. Learning and improvising are done until best accuracy is obtained, which is also called model tuning. Similarly, for the testing data, the dataset goes through pre- processing, feature measurement and finally evaluating the result. The testing dataset checks the performance of the model. With lots of improvisation, a classification model with acceptable accuracy is obtained. The results of decision tree and random forest are evaluated through different performance measures such as f1 score, recall, accuracy and precision. By analyzing different data entered by the user such as: gender, hypertension level, smoking status etc. we predict brain stroke.

3.3 PREDICTIVE MARKERS

The predictive markers, derived from various medical, lifestyle, and physiological factors, serve as valuable indicators to assess the likelihood of a brain stroke. In this section, we explored the predictive markers for brain stroke, shedding light on the significance of these markers and their role in enhancing stroke prediction models. The factors that have been identified for brain stroke prediction in our dataset are age, gender, heart disease, average glucose level, body mass index, hypertension, ever married, work type, residence type and smoking status.

Age:

The older the person gets, he or she is more likely to have stroke. The chance of having a stroke about doubles every 10 years after age 55. Although stroke is common among older adults, many people younger than 65 years also have strokes. Experts think, nowadays younger people are having more strokes because more young people have obesity, high blood pressure, and diabetes.

Gender:

Stroke is more common in women than men, and women of all ages are more likely than men to die from stroke. Pregnancy and use of birth control pills increases the risk of stroke for women.

Heart disease:

People having heart disease tend to have higher risk of stroke in compared to those who do not have if proper care is not taken.

Average glucose level:

High glucose levels can damage the body’s blood vessels, increasing the chance of stroke. A fasting blood glucose (sugar) level of 126 milligrams per deciliter (mg/dL) or higher is dangerous. People who have diabetes are 2 times as likely to have a stroke compared to people who do not have diabetes.

Body mass index:

The studies estimate that each unit increase in body mass index (BMI) increases the risk of stroke by 5 percent. Men with a BMI of 30 or higher are found to be twice as likely to have a stroke compared with men who had a BMI of less than 23.

Ever married:

Singles have significantly increased likelihood of having stroke than married people. Also widowed or divorced people have a somewhat lower risk of getting stroke than those who have never been married before.

Residence type:

It was found that rural residence was associated with higher rates of stroke in both primary prevention (adjusted hazard ratio[aHR] for stroke 1.05, 95% of CI) and secondary prevention ([aHR] for stroke 1.11, 95% of CI) [7].

Work type:

People who work on private companies are most likely to have strokes, whereas the ones having government jobs have less chances of suffering from stroke. Likewise, for self employed, it is between the private companies and government job holder.

Hypertension:

Hypertension or high blood pressure is a significant risk factor for the development of brain strokes. The relationship between hypertension and brain strokes is complex, and there is no specific blood pressure level that directly causes a stroke. However, prolonged or uncontrolled high blood pressure significantly increases the risk of stroke occurrence.

Smoking status:

Smoking is a significant risk factor for stroke, as it increases the likelihood of blood clots forming in the blood vessels. Smokers are 1.5 to 2.5 times more likely to have an ischemic stroke and are 2.5 times more likely to have hemorrhagic stroke compared to non-smokers.

3.4 ALGORITHM

3.4.1 Random Forest Classifier

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in machine leaning. It is based on the concept of ensemble learning which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

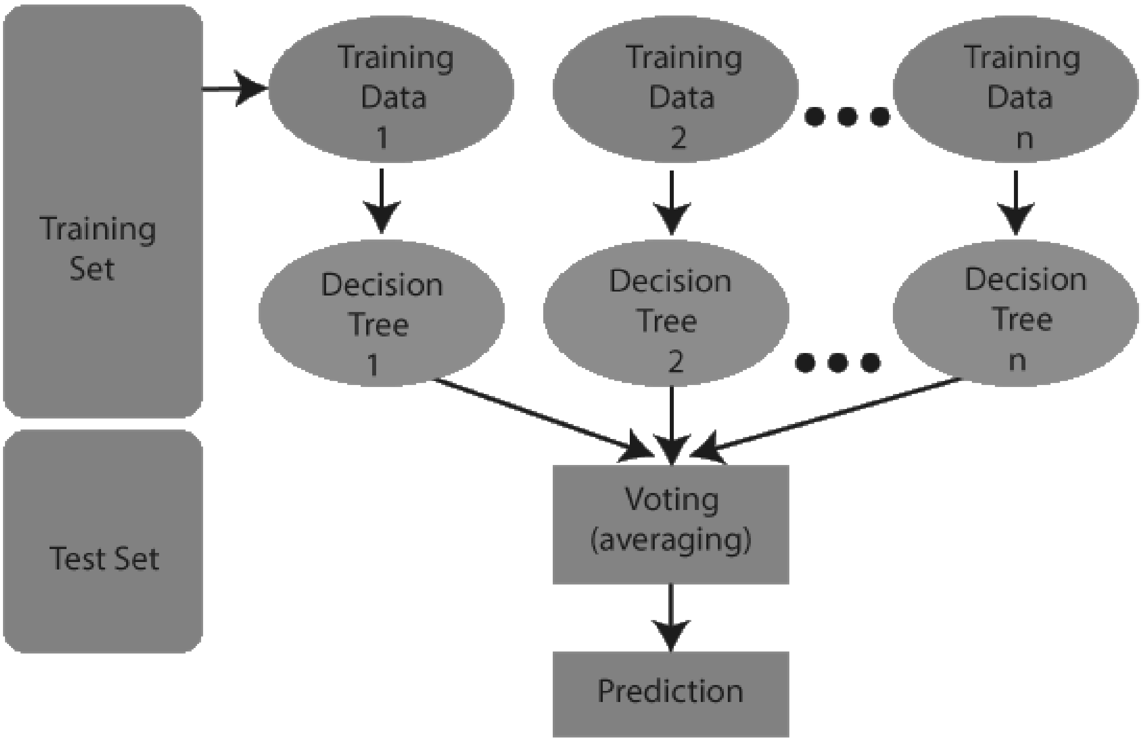


Figure 3.3 Random Forest [8]

Random forest combines the simplicity of decision tree with flexibility which results in a vast improvement of accuracy. A random forest produces good predictions that can be understood easily. Random forest can be a valuable algorithm for stroke prediction due to its ability to handle a combination of categorical and numerical features, capture complex relationships, and provide feature importance analysis. Random Forest works in two phases: first is to create the random forest by combining N decision trees, and second is to make predictions for each tree created in the first phase.

The working process can be explained in the steps below:

Step 1: Select random K data points from the training set.

Step 2: Build the decision trees associated with the selected data points (Subsets).

Step 3: Choose the number N for decision trees that you want to build.

Step 4: Repeat Step 1 and 2.

Step 5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

3.4.2 Decision Tree Classifier

Decision tree classifier is a supervised learning approach which builds tree-like model of decision based on the features of the input data to predict the class or category of the target variable. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf nodes represents the outcome. In this classification algorithm we have two type of nodes, Decision node and Leaf node. Decision node are used to make decision and have multiple branches, Leaf node are the output of those decision and do not contain further branches.

In a decision tree, for predicting the class of the give dataset, the algorithm starts from the root node of the tree. This algorithm compares the values of the root attribute with the real dataset attribute and based, on the comparison, follows the branch and jumps to the next node. For the next node the algorithm again compares the attribute value with the other sub-node and move further. It continues the process until it reaches the leaf node of the tree. The process can be understood more clearly from the algorithm below:

Step 1: Begin the tree with the root node, say S, which contain the complete dataset.

Step 2: Find the best attribute in the dataset using Attribute Selection Measure (ASM).

Step 3: Divide the S dataset into subset that contains the possible values for the best attribute.

Step 4: Generate the decision tree node, which contains the best attribute.

Step 5: Recursively make new decision tree using the subsets of the dataset created in step 3. Continue this process until a stage is reached where we cannot classify the nodes further.

While implementing a Decision tree, the main problem arises on how to select the best attribute for the root node and the sub node. In order to solve this problem there is a technique called as Attribute selection measure (ASM). Using this measure we can easily select the attribute for the nodes of tree.

The popular techniques for ASM are:

1. Information Gain

Information Gain is based on the concept of entropy and information content from information theory. Entropy is a metric to measure the impurity in a given attribute. It specifies randomness in data. According to the value of information gain, we split the node and build the decision tree. A decision tree algorithm always tries to maximize the value of information gain, and node/attribute having the highest information gain is split first. It is calculated using the formula:

Information Gain= Entropy(S)- [(Weighted Avg) *Entropy(each feature)]

Where, S is the total number of samples.

2. Gini Index

Gini index is a measure of impurity or purity used while creating a decision tree. An attribute with low gini index should be preferred as compared to the high gini index. It can be calculated using the formula:

$$\text{Gini Index} = 1 - \sum_j P_j^2$$

3.5 COMPARISON OF ALGORITHMS

	Decision tree algorithm	Random forest algorithm
Accuracy	0.948	0.968
F1 score	0.947	0.966
Recall	0.945	0.940
Precision	0.949	0.993

Table 1.1 Comparision of algorithms

The above table shows the comparison between the decision tree classifier and random forest classifier algorithms based on performance metrics. The decision tree algorithm exhibited an accuracy of 0.948, while the random forest algorithm gave a higher accuracy of 0.968. In terms of F1 score, the decision tree achieved 0.947, while the random forest demonstrated a slightly higher score of 0.966. Both algorithms showed strong recall, with the decision tree at 0.945 and the random forest at 0.940. Notably, the precision of the decision tree was 0.949, whereas the random forest algorithm excelled in precision with a score of 0.993. These results suggest that the random forest algorithm, in this context, displays a notable advantage in accuracy and precision compared to the decision tree algorithm, making it a potentially preferable between choices.

3.6 SYSTEM FLOWCHART

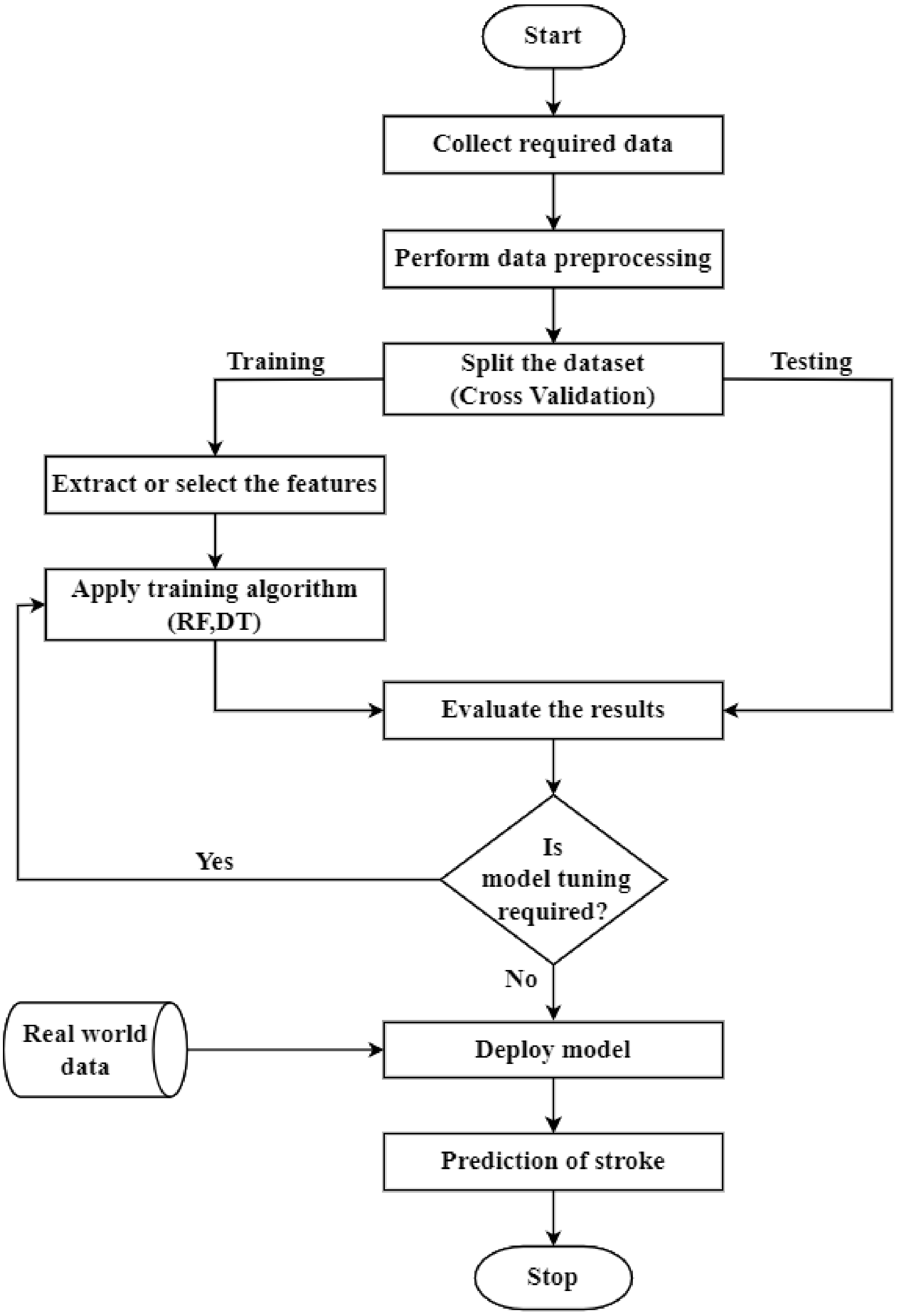


Figure 3.4 Flowchart for stroke prediction

3.7 UML Diagrams

3.7.1 Use Case Diagram

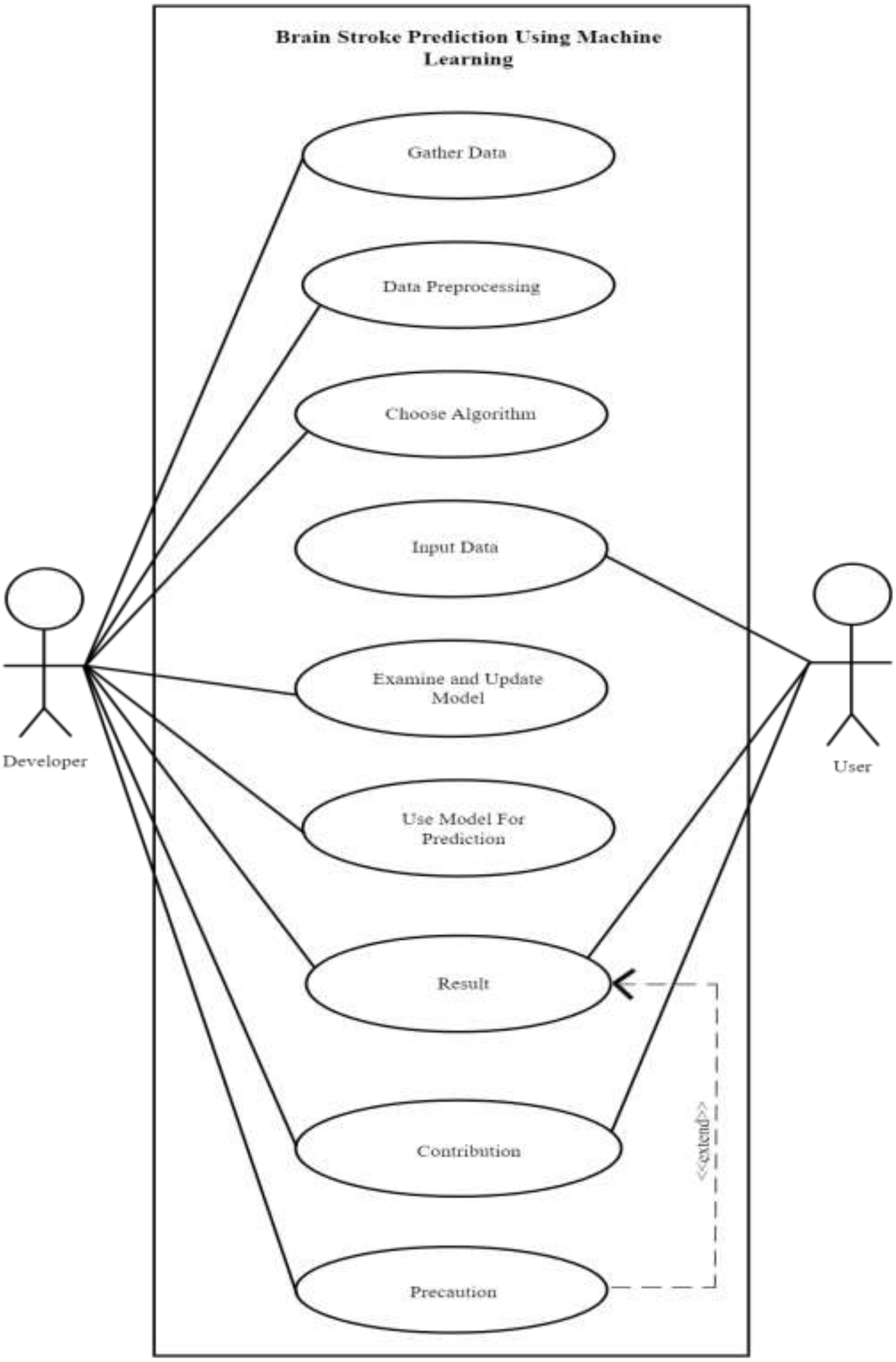


Figure 3.5 Use Case Diagram

Figure 3.5 represents a use case diagram consisting of two actors, Developer and User. The horizontally shaped ovals represent the different use cases that represent the functionality of the system. Also, the line between actors and use cases are the association which means that which actors are associated with which use cases. The use cases are gathering data, data pre-processing, choose algorithm, input data, examine and update the model, use model for prediction, result, conclusion and precaution.

The developer gathers data followed by data preprocessing and chooses the suitable algorithm. The user can input data manually, see the result and also find out percentage of how much each factor are contributing that is feature contribution; for the occurrence of stroke. There is an extend relation between precaution and result that indicates the precaution is only displayed when the result predicts positive. Also, the preventive measures are displayed if there is possibility of suffering from stroke.

3.7.2 Sequence Diagram

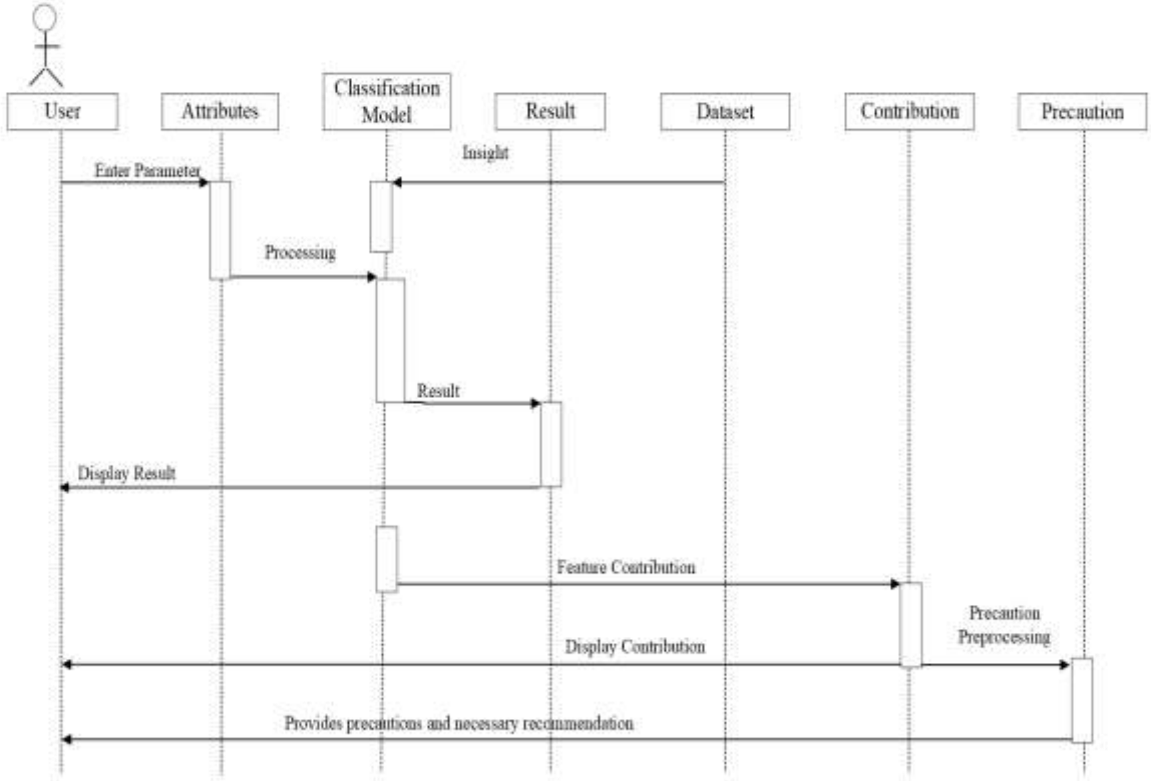


Figure 3.6 UML Sequence Diagram

UML sequence diagram is the diagram that shows how objects interact with each other through message passing with reference to time where each object is represented by a lifeline. It visualizes the sequence of messages passed between different objects or components within a system over a period of time. Here, in above figure (Figure 3.6) there are seven lifelines, User, Attributes, Classification Model, Result, Dataset, Contribution and Precaution. Each of these lifelines represents an individual participant in the interaction. The thin rectangle on the lifeline represents the period during which an element is performing an operation (i.e. activation time) and the call messages defines a particular communication between lifelines of an interaction.

The system operates by utilizing a dataset to train a classification model. When a user inputs parameters such as age, hypertension, BMI, heart disease, and other relevant factors, the system processes this information. This involves passing the user's input through the trained classification model, which then calculates a result. The result not only informs the user about the likelihood of a stroke occurrence but also provides insight into how much each feature (age, hypertension, BMI, heart disease, etc.) contributes to this outcome. This information is crucial as it helps users understand the factors influencing the predicted result.

Furthermore, if there is a possibility of a stroke occurrence based on the user's input and the model's prediction, the system takes proactive measures. It displays precautions along with necessary recommendations to the user. These precautions could include lifestyle changes, medical interventions, or other actions aimed at reducing the risk of stroke.

3.8 TOOLS USED

3.8.1 Python

Python is an interpreted high-level general purpose programming language created by Guido van Rossum, and released in 1991. It supports multiple programming paradigms, including structured, object oriented and functional programming. Python’s design philosophy emphasizes code readability with its notable use of significant indentation. Its language constructs as well as its object-oriented approach aims to help programmers write clear, logical code for small and large-scale projects.

3.8.2 NumPy

NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, Fourier transform, and matrices. NumPy was created in 2005 by Travis Oliphant. NumPy stands for Numerical Python. It supports large, multi-dimensional arrays and matrices along with a large collection of high level mathematical functions to operate on the arrays.

3.8.3 Pandas

Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data. The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008. Pandas allows us to analyze big data and make conclusions based on statistical theories. Pandas are also able to delete rows that are not relevant, or contain wrong values, like empty or NULL values. This is called cleaning the data.

3.8.4 Seaborn

Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures. Seaborn helps to explore and understand the data. Its plotting functions operate on data frames and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots. Its dataset-oriented, declarative API lets us focus on what the different elements of our plots mean, rather than on the details of how to draw them.

3.8.5 Jupyter Notebook

The Jupyter Notebook is an open source web application that we can use to create and share documents that contain live code, equations, visualizations, and text. The name, Jupyter, comes from the core supported programming languages that it supports: Julia, Python, and R. Jupyter ships with the IPython kernel, which allows us to write your programs in Python, but there are currently over 100 other kernels that we can also use.

3.8.6 React

React is an open-source JavaScript library that is primarily used for building user interfaces (UIs) or user interface components for web applications. It was developed and is maintained by Facebook. React was designed to make the process of building interactive and dynamic user interfaces more straightforward and efficient.

3.8.7 Flask

Flask is a lightweight and web framework for Python designed to make it easy to develop web applications quickly and with minimal code. It is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries. Flask provides the essentials needed for building web applications and leaves other components to be integrated as needed. Flask is a popular choice for developers who prefer a minimalistic framework and want the freedom to choose components according to their project's needs.

3.8.8. Matplotlib

Matplotlib is plotting library for the Python programming language and its numerical mathematics extension NumPy. Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. It is a visualization library for 2D plot of arrays. It consists of several plots like bar plot, scatter plot, etc.

3.8.9. Scikit-learn

Scikit-learn is a library which provides a range of supervised and unsupervised machine learning algorithms like logistic regression, random forest, etc. With scikit-learn, we can use pre-written code to do things like predicting outcomes based on data, finding patterns in data, or making decisions. Scikit-learn makes it simpler to do machine learning tasks without having to write everything from scratch.

3.8.10. GridSearchCV

GridSearchCV is a helper tool in scikit-learn that helps to find the best parameters for our machine learning models. It takes a bunch of different values for a parameter and then tries out all possible combinations of them. It trains and evaluates the model with each combination using a technique called cross-validation, where the data is split into multiple parts for training and testing.

3.9 VERIFICATION AND VALIDATION

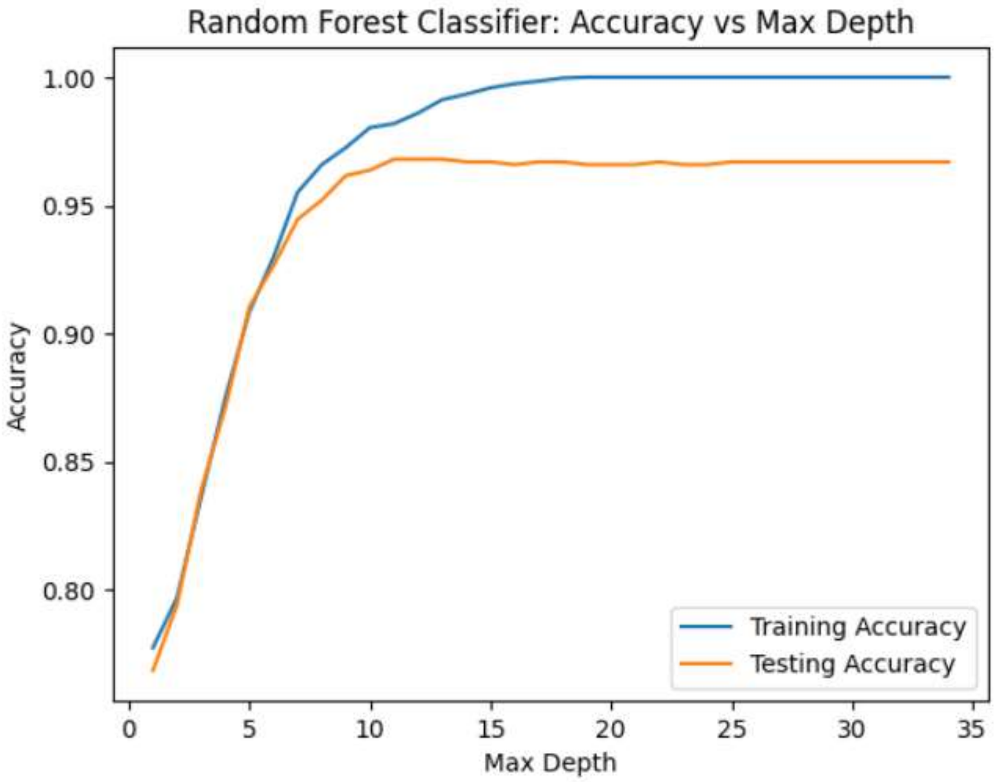


Figure 3.7: Accuracy Graph of Training and Validation

CHAPTER 4: EPILOGUE

4.1 Result

The GUI has been successfully implemented from which the attributes values can be passed manually and we get the prediction for whether the user has the possibility of having stroke or not. If the user has the possibility of having stroke, the precaution is displayed according to the contributing factor i.e. bmi, average glucose level, hypertension, heart disease, age etc. which is generated from the calculation of feature contribution of each parameter. To make the site more presentable we have employed the feature provided by the react library called parallax. This feature adds depth and dimension to the interface, creating an engaging and immersive browsing experience for users. For the accuracy comparison, it was found that the accuracy with Decision tree classifier was 94.8% and Random Forest was 97%. We also used different performance metrics like F1- score, recall and precision for the comparison. For our project, out of the two algorithms taken into implementation, Random Forest was found to be the best algorithm for all the stroke data.

4.2 Conclusion

In conclusion, the development of an accurate and reliable model for predicting stroke occurrence represents a significant step towards addressing the burden of stroke-related mortality. Through the utilization of various data our project aims to empower individuals and healthcare providers with early detection tools and preventive strategies customized to individual who are at risk of stroke.

By analyzing factors such as age, gender, bmi and lifestyle choices, our model seeks to identify those at high risk of stroke, enabling timely detection and reducing the severity of potential stroke events. The feature contribution along with percentage contribution of each attribute will be displayed to the user. Furthermore, our project helps analyze the health parameters and provision of personalized recommendations for lifestyle modifications and other preventive measures. By integrating these recommendations into clinical practice, we aim to not only reduce the incidence of stroke but also improve overall population health outcomes.

Hence, StrokeOpus is a platform which can be used by a general public in order to check if the person has possibility of having stroke. If the user is likely to have stroke, then the precaution along with the feature contribution of each attribute from which the user can identify the factor affecting the result.

4.3 Future Enhancement

We aim to implement the following enhancements in the future:

- Hardware implementation using external camera for image processing.
- Increase the number of attributes used in prediction.

REFERENCES

[1] Centers for Disease Control and Prevention, “About Stroke”, May 04, 2023 [Online] Available: https://www.cdc.gov/stroke/about.htm. [Accessed Jun 21, 2023]

[2] M Pyakurel, S Bhattarai, B Joshi, R P Koju, A Shrestha, “Burden of Stroke in Nepal: Findings from Global Burden of Disease Dataset 2017”,. Available: https://pubmed.ncbi.nlm.nih.gov/35526132/ [Accessed Jun 21, 2023]

[3] P. Warlow, “Epidemiology of stroke,” *The Lancet*, vol. 352, no. 3, pp. 1–4, 1998.

[4] Donkor E. S. (2018). Stroke in the 21st Century: A Snapshot of the Burden, Epidemiology, and Quality of Life. Stroke research and treatment, 2018, 3238165. Available: https://doi.org/10.1155/2018/3238165

[5] Tianyu Liu, Wenhui Fan, Cheng Wu(2019), A hybrid machine learning approach to cerebral stroke prediction based on imbalanced medical dataset, Artificial Intelligence in Medicine, Volume 101, https://doi.org/10.1016/j.artmed.2019.101723

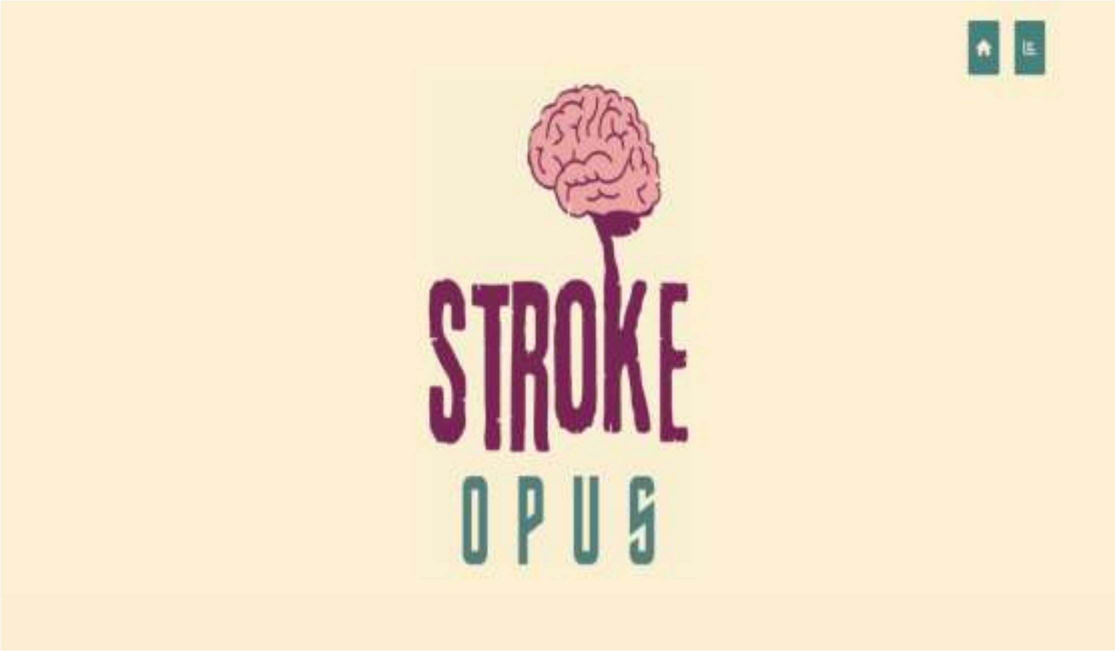
[6] National Institute Of Neurological And Stroke Disorder, “Stroke”. [Online]. Available: https://www.ninds.nih.gov/Disorders/All-Disorders/Stroke-Information-Page [Accessed Jun 21, 2023]

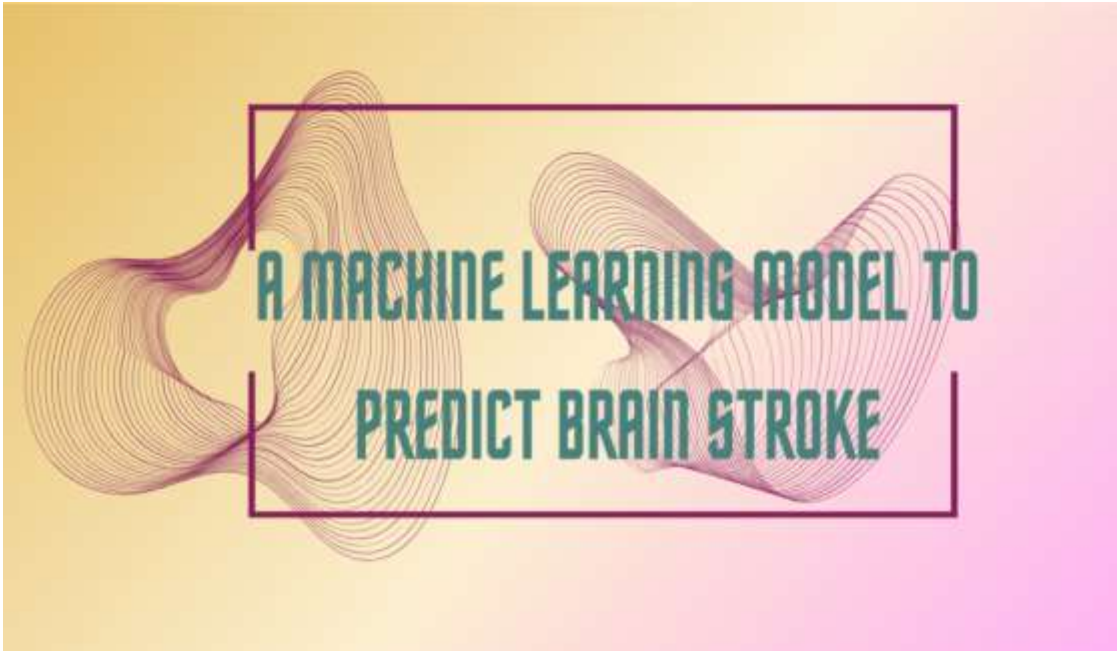
[7] Circulation: Cardiovascular quality and outcomes “Rural-Urban Differences in Stroke Risk Factor, Incidence and Mortality in People with and without prior Stroke” Available: https://doi.org/10.1161/CIRCOUTCOMES.118.004973

[8] JavaTpoint,” Decision tree classification algorithm,’’ [Online]. Available: https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm. [Accessed 2021]

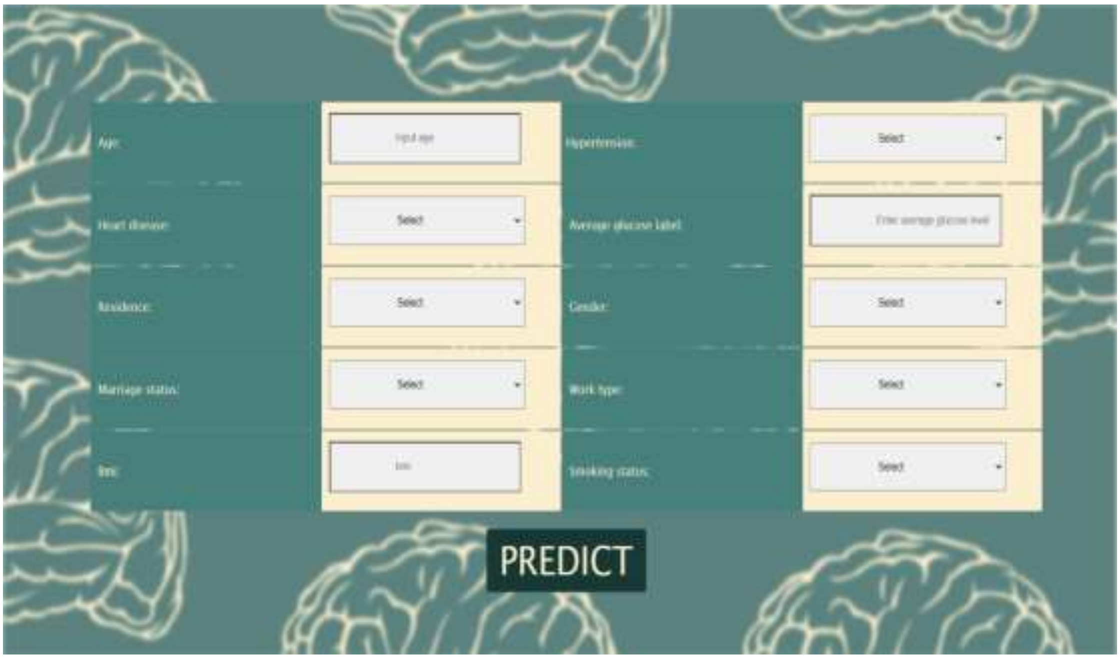
SCREENSHOTS

A. Home Page



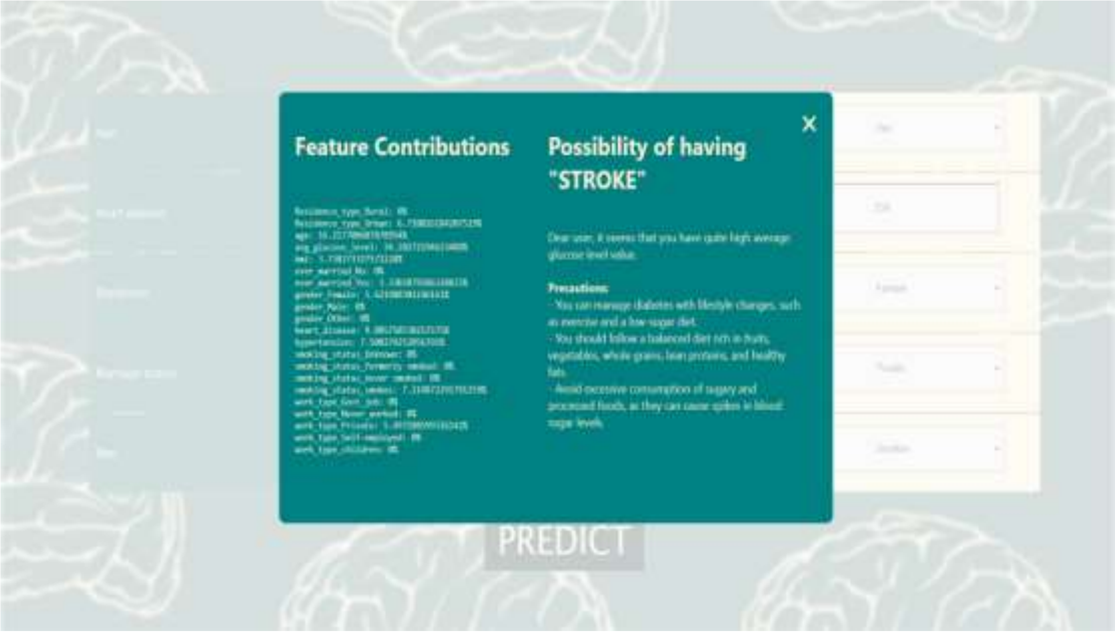


B. User Input



Age:	Input age	Hypertension:	Select
Heart disease:	Select	Average glucose label:	Enter average glucose level
Residence:	Select	Gender:	Select
Marriage status:	Select	Work type:	Select
bmi:	bmi	Smoking status:	Select

C. Messages displayed based on user's input



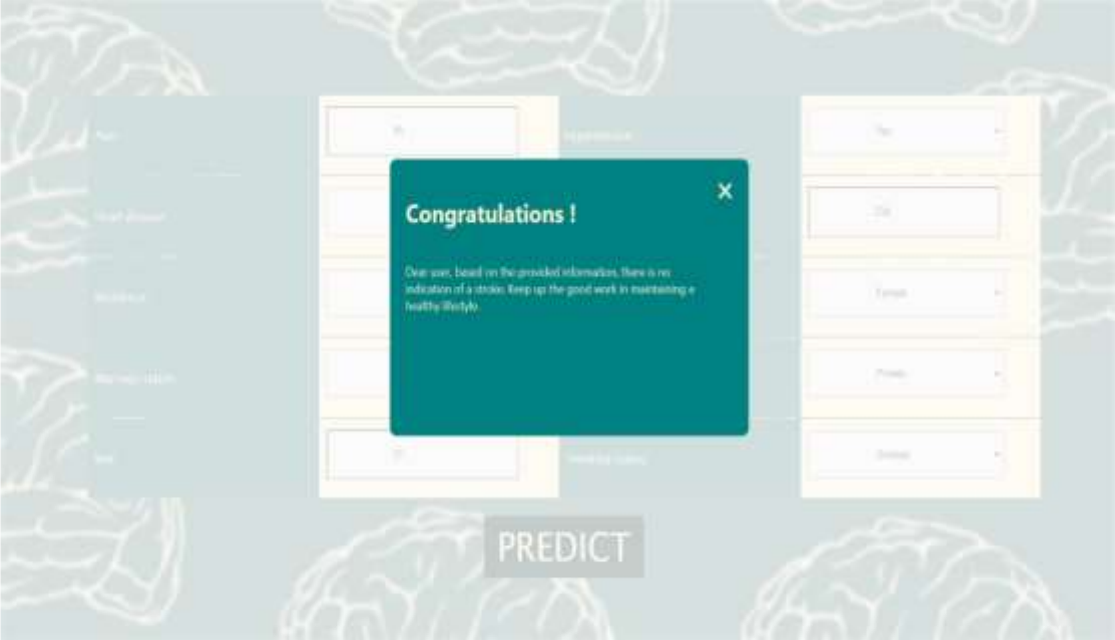
Feature Contributions

```
Residence_type_Rural: 0%
Residence_type_Urban: [illegible]
age: [illegible]
avg_glucose_level: [illegible]
bmi: [illegible]
ever_married_No: 0%
ever_married_Yes: [illegible]
gender_Female: [illegible]
gender_Male: 0%
gender_Other: 0%
heart_disease: [illegible]
hypertension: [illegible]
smoking_status_Unknown: 0%
smoking_status_formerly smoked: 0%
smoking_status_never smoked: 0%
smoking_status_smokes: [illegible]
work_type_Govt_job: 0%
work_type_Never_worked: 0%
work_type_Private: [illegible]
work_type_Self-employed: 0%
work_type_children: 0%
```

Possibility of having "STROKE"

Dear user, it seems that you have quite high average glucose level value.

Precautions:
- You can manage diabetes with lifestyle changes, such as exercise and a low-sugar diet.
- You should follow a balanced diet rich in fruits, vegetables, whole grains, lean proteins, and healthy fats.
- Avoid excessive consumption of sugary and processed foods, as they can cause spikes in blood sugar levels.



Congratulations !

Dear user, based on the provided information, there is no indication of a stroke. Keep up the good work in maintaining a healthy lifestyle.