A

Major Project On

### STROKE RISK PREDICTION USING MACHINE

**LEARNING ALGORITHMS**

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

VOBILISHETTY SURAJ KUMAR (207R1A05B8)

Under the Guidance of

#### K.SHILPA

(Assistant Professor)



#### DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**CMR TECHNICAL CAMPUS UGC AUTONOMOUS**

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New

Delhi) Recognized Under Section 2(f)&12(B) of the UGC Act.1956, Kandlakoya(V), Medchal Road, Hyderabad-501401.

**2020-2024**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



### CERTIFICATE

This is to certify that the project entitled **“STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS”** being submitted by **VOBILISHETTY SURAJ KUMAR (207R1A05B8)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by them under our guidance and supervision during the year 2023-24.

The results embodied in this project have not been submitted to any other University or Institute for the award of any degree or diploma.

**K.SHILPA Dr. A. RajiReddy**

(Assistant Professor) DIRECTOR

INTERNALGUIDE

**Dr. K. Srujan Raju EXTERNALEXAMINER**

HOD

**Submitted for vivavoce Examination held on**

#### ACKNOWLEDGEMENT

A part from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. I take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

I take this opportunity to express my profound gratitude and keep regard to my guide **K. SHILPA**, Assistant Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which I was about to embark.

I also take this opportunity to express a deep sense of gratitude to the Project Review Committee (PRC) **G. Vinesh Shanker, Dr. J. Narasimharao, Ms.**

**K. Shilpa, & Dr. K. Maheswari** for their cordial support, valuable information and guidance, which helped me in completing this task through various stages.

I am also thankful to **Dr. K. Srujan Raju,** Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

I am obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course of this project. I also express our sincere gratitude to Sri.**Ch. Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. I am grateful for their constant support and help.

Finally, I would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. I sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

**VOBILISHETTY SURAJ KUMAR (207R1A05B8)**

#### ABSTRACT

The project focuses on leveraging machine learning algorithms to detect stroke warning symptoms early, crucial for prompt intervention and treatment. Through the use of classifiers like J48, ANN, KNN, Random Forest, and Naïve Bayes. This research holds promise for improving stroke detection and intervention, potentially saving lives through early identification of symptoms Random Forest stands out as the top performer in predicting strokes with high accuracy. The user-friendly website makes accessing the stroke prediction model easy for medical professionals and patients. Stroke risk prediction using machine learning algorithms has gained significant attention due to its potential in early identification and prevention of stroke occurrences. In this project, the propose a novel approach for stroke risk prediction leveraging various machine learning algorithms. Then begin by collecting comprehensive demographic, clinical, and lifestyle data from a large cohort of individuals. These data include variables such as age, gender, blood pressure, cholesterol levels, smoking status, and family history of stroke.

Next, the preprocess data to handle missing values, normalize features, and encode categorical variables. Subsequently, the employ several machine learning algorithms like Naïve bayes, K-Nearest neighbor, Random forest, J48, ANN, to build predictive models. These models are trained on the dataset and evaluated using metrics such as accuracy, precision, recall, and F1 score.

Among the various algorithms tested, random forest emerges as the most promising model for stroke risk prediction. Random forest excels in handling complex interactions between features and provides robust predictions even in the presence of noisy or correlated variables. By leveraging an ensemble of decision trees, random forest effectively captures non-linear relationships and feature importance, enhancing the interpretability of the model.

|  |  |  |
| --- | --- | --- |
| **FIGURENO** | **FIGURENAME** | **PAGENO** |
| Figure3.1 | Project Architecture for Stroke risk prediction using machine learning algorithms | 10 |
| Figure3.2 | Use Case Diagram for Stroke risk prediction using machine learning algorithms | 12 |
| Figure3.3 | Class Diagram for Stroke risk prediction using machine learning algorithms | 13 |
| Figure3.4 | Sequence diagram for Stroke risk prediction using machine learning algorithms | 14 |
| Figure3.5 | Activity diagram for Stroke risk  Prediction using machine learning algorithms | 15 |

**LIST OF SCREENSHOTS**

|  |  |  |
| --- | --- | --- |
| **SCREENSHOT NO.** | **SCREENSHOT NAME** | **PAGENO** |
| Screenshot 5.1 | Upload Stroke Dataset | 26 |
| Screenshot 5.2 | Dataset Preprocessing & Features Selection | 26 |
| Screenshot 5.3 | Train Naïve Bayes Algorithm | 27 |
| Screenshot 5.4 | Train J48 Algorithm | 27 |
| Screenshot 5.5 | Run KNN Algorithm | 28 |
| Screenshot 5.6 | Run Random Forest Algorithm | 28 |
| Screenshot 5.7 | Command prompt | 29 |
| Screenshot 5.8 | Run ANN Algorithm | 29 |
| Screenshot 5.9 | Comparison Graph | 30 |
| Screenshot 5.10 | High Accuracy Graph | 30 |
| Screenshot 5.11 | Train Test Data | 31 |
| Screenshot 5.12 | Prediction of Disease | 31 |

|  |  |
| --- | --- |
| **ABSTRACT**  **LIST OF FIGURES**  **LIST OF SCREENSHOTS**  **1.**[**INTRODUCTION**](#_bookmark0) | **i ii iii**  [**1**](#_bookmark0) |
| [PROJECTSCOPE](#_bookmark1) | [1](#_bookmark1) |
| [PROJECTPURPOSE](#_bookmark2) | [1](#_bookmark2) |
| [PROJECTFEATURES](#_bookmark3) | [1](#_bookmark3) |
| **2.**[**SYSTEMANALYSIS**](#_bookmark4) | [**3**](#_bookmark4) |
| [PROBLEMDEFINITION](#_bookmark5) | [3](#_bookmark5) |
| [EXISTINGSYSTEM](#_bookmark6) | [3](#_bookmark6) |
| LIMITATIONSOFTHEEXISTINGSYSTEM | 4 |
| LITERATURE SURVEY | 4 |
| PROPOSEDSYSTEM | 6 |
| ADVANTAGESOFPROPOSEDSYSTEM | 7 |
| [FEASIBILITYSTUDY](#_bookmark7) | [7](#_bookmark7) |
| [ECONOMICFEASIBILITY](#_bookmark8) | 8 |
| [TECHNICALFEASIBILITY](#_bookmark9) | [8](#_bookmark9) |
| SOCIALFEASIBILITY | 8 |
| [HARDWARE&SOFTWAREREQUIREMENTS](#_bookmark10) | [9](#_bookmark10) |
| [HARDWAREREQUIREMENTS](#_bookmark11) | [9](#_bookmark11) |
| [SOFTWAREREQUIREMENTS](#_bookmark12) | [9](#_bookmark12) |
| **3.**[**ARCHITECTURE**](#_bookmark13) | **10** |
| [PROJECTARCHITECTURE](#_bookmark14) | [10](#_bookmark14) |
| [DESCRIPTION](#_bookmark15) | 11 |
| [USE CASE DIAGRAM](#_bookmark16) | 12 |
| [CLASSDIAGRAM](#_bookmark17) | 13 |
| [SEQUENCE DIAGRAM](#_bookmark18) | [1](#_bookmark18)4 |
| [ACTIVITYDIAGRAM](#_bookmark19) | [1](#_bookmark19)5 |
| **4. IMPLEMENTATION** | **16** |
| SAMPLECODE | 16 |

##### RESULTS 26

6.[TESTING 32](#_TOC_250009)  [INTRODUCTION TO TESTING 3](#_TOC_250008)2

[TYPES OF TESTING 3](#_TOC_250007)2

UNIT TESTING 32

[INTEGRATION TESTING 3](#_TOC_250006)2

[FUNCTIONAL TESTING 3](#_TOC_250005)3

[TEST CASES 36](#_TOC_250004)

[CLASSIFICATION 3](#_TOC_250003)6

7. CONCLUSION & FUTURE SCOPE 36

PROJECT CONCLUSION 37

[FUTURE SCOPE](#_TOC_250002) 37

##### BIBLIOGRAPHY 38

[REFERENCES](#_TOC_250001) 39

[GITHUB LINK](#_TOC_250000)  40

# 

# 1. INTRODUCTION

### INTRODUCTION

STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS

#### PROJECT SCOPE

The scope of this project revolves around developing a machine learning model for predicting stroke risk. This involves collecting and analyzing relevant medical data, selecting appropriate features, training various machine learning algorithms, and ultimately deploying a predictive model capable of assessing stroke risk accurately.

.

#### PROJECT PURPOSE

The primary objective of this project is to create are liable tool that can assist healthcare professionals in identifying individuals who are at higher risk of experiencing a stroke. By leveraging machine learning techniques, the aim is to enhance early detection and prevention strategies, there by potentially reducing the incidence and severity of strokes**.**

#### PROJECT FEATURES

* Data Collection and Preprocessing: Gather medical data including demographic information, lifestyle factors, medical history, and biometric measurements. This data will be cleaned, normalized, and processed to ensure consistency and accuracy.

1

CMRTC

STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS

* Feature Selection: Utilize techniques such as correlation analysis, feature importance ranking, and domain expertise to select the most relevant features contributing to stroke risk prediction. Features may include age, gender, blood pressure, cholesterol levels, smoking status, and medical history of conditions like hypertension and diabetes.
* Model Training and Evaluation: Experiment with various machine learning algorithms including J48, Naïve bayes, KNN, ANN, and random forest. Train these models using labeled data and evaluate their performance using metrics such as accuracy, precision, recall, and F1-score through cross- validation.
* Hyper parameter Tuning: Fine-tune the parameters of the selected algorithms to optimize their performance and generalization capabilities. Techniques like grid search or randomized search can be employed for hyper parameter optimization.
* Model Interpretability: Ensure that the developed model is interpretable, allowing healthcare professionals to understand the factors contributing to an individual's stroke risk.
* Deployment: Deploy the trained model in a user-friendly interface accessible to healthcare providers. This could be in the form of a web application or integration with existing electronic health record systems.

2

CMRTC

## SYSTEM ANALYSIS

#### SYSTEMANALYSIS

#### SYSTEM ANALYSIS

System analysis is a critical phase in the development lifecycle where the current system or problem domain is thoroughly examined. The primary goal is to gather requirements, assess feasibility, and model the system. This phase involves engaging with stake holders to understand their needs, conducting a feasibility study to evaluate project viability, creating models to represent system components, and identifying potential risks that may impact the project's success.

#### PROBLEM DEFINITION

Problem definition is the initial step in system development, focusing on understanding and articulating the challenges that the new system aims to address. It involves identifying the problem, setting clear objectives aligned with organizational goals, defining the scope and boundaries of the system, and analyzing constraints. By clearly defining the problem and project objectives, this phase provides a roadmap for the subsequent stages of system development, ensuring a focused and effective approach.

#### EXISTING SYSTEM

Stroke is a significant health concern worldwide, contributing to high morbidity and mortality rates. Early identification of individual satrisk of stroke is crucial for preventive measures and timely intervention. In recent years, machine learning (ML) techniques have shown promise in predicting stroke risk by leveraging various risk factors and biomarkers.

This project aims to develop a stroke risk prediction system using

Machine learning algorithms. By analyzing a comprehensive set of patient data, medical history, lifestyle factors the proposed system will provide personalized risk assessments for individuals. Such predictive models can aid healthcare professionals in identifying high-risk individuals, enabling targeted interventions and preventive strategies to mitigate the incidence of stroke.

##### DISADVANTAGES OF EXISTING SYSTEM

* Data loss
* No security
* Less efficiency
* Less accuracy
* Time consuming

##### LITERATURE SURVEY

In recent years, machine learning algorithms have gained significant traction in predicting stroke risk due to their ability to effectively handle large data sets and extract complex patterns from various risk factors. A literature survey reveals a plethora of studies employing diverse machine learning techniques for stroke risk prediction. Among these algorithms, Random Forest stands out as a particularly promising choice due to its robustness and ability to handle both categorical and continuous data while mitigating overfitting issues.

Random Forest, a popular ensemble learning method, constructs a multitude of decision trees during training and outputs them ode of the classes (classification) or the mean prediction (regression) of individual trees. Its inherent ability to handle high-dimensional datasets and non linear relationships between predictors makes it suitable for stroke risk prediction tasks. Several studies in the literature have demonstrated the efficacy of Random Forest in accurately identifying individuals at risk of stroke based on a variety of input features.

Moreover, Random Forest has been leveraged in conjunction with advanced feature engineering methods such as feature importance ranking and partial dependence plots to gain insights into the underlying mechanisms contributing to stroke risk. This not only aids in the development of more effective predictive models but also enhances our understanding of the complex interplay between different risk factors associated with stroke occurrence.

In summary, Random Forest emerges as a powerful tool in the realm of stroke risk prediction, offering a balance between predictive performance, interpretability, and scalability. Its ability to handle heterogeneous data types and capture nonlinear relationships makes it well-suited for addressing the multi faceted nature of stroke risk assessment, there by holding significant promise for improving patient outcomes through proactive risk management strategies.In order to get required knowledge about various concepts related to the present analysis existing literature were studied. Some of the important conclusions were made through those are listed below.

“Computer Methods and Programs in Biomedicine” - Jae–woo Lee, Hyunsun Lim, Dong-wook Kim, Soon-ae Shin, Jinkwon Kim, Bora Yoo, Kyunghee Cho – The Purpose of this project was Calculation of 10-year stroke prediction probability and classifying the user's individual probability of stroke into five categories.

“Probability of Stroke: A Risk Profile from the Framingham Study” - Philip A. Wolf, MD; Ralph B.D' Agostino, PhD, Albert J. Belanger,MA; and William B. Kannel, MD - In this paper, A health risk appraisal function has been developed for the prediction of stroke using the Framingham Study cohort.

“Development of an Algorithm for Stroke Prediction: A National Health Insurance Database Study” -Min SN, Park SJ, Kim DJ, Subramaniyam M, Lee KS – In this research, this paper aimed to derive a model equation for developing a stroke pre- diagnosis algorithm with the potentially modifiable risk factors.

“Stroke prediction using artificial intelligence”- M. Sheetal Singh, Prakash Choudhary - In this paper, Here, decision tree algorithm is used for feature selection process, principle component analysis algorithm is used for reducing the dimension and adopted back propagation neural network classification algorithm, to construct a classification model.

“Medical software user interfaces, stroke MD application design (IEEE)” Elena Zamsa-The article presents the design of an application interface for associated medical data visualization and management for neurologists in a stroke clustering and prediction system called Stroke MD.

“Focus on stroke: Predicting and preventing stroke” Michael RegnierThis paper focuses on cutting-edge prevention of stroke.

“Effective Analysis and Predictive Model of Stroke Disease using Classification Methods”-A.Sudha, P.Gayathri,N.Jaisankar- This project, principle component analysis algorithm is used for reducing the dimensions and it determines the attributes involving more towards the prediction of stroke disease and predicts whether the patient is suffering from stroke disease or not.

“Deep learning algorithms for detection of critical findings in head CT scans: a retrospective study” - Rohit Ghosh, Swetha Tanamala, Mustafa Biviji, Norbert G Campeau, Vasantha Kumar Venugopal - In this project Non-contrast head CT scan is the current standard for initial imaging of patients with head trauma or stroke symptoms. This article aimed to develop and validate a set of deep learning algorithms for automated detection.

#### PROPOSED SYSTEM

In our proposed system for stroke risk prediction using machine learning algorithms, we aim to leverage various features and data points to create a predictive model. This model will analyse individual patient data to assess their risk of experiencing a stroke within a certain timeframe.

Initially, we'll gather a comprehensive dataset containing relevant medical information such as demo graphics, medical history, lifestyle factors, and any existing health conditions. This dataset will serve as the basis for training our machine learning algorithms.

We'll employ various machine learning algorithms such as J48, Naïve bayes, KNN, ANN and random forest. These algorithms will undergo training on the dataset to learn patterns and relationships between different variables and stroke occurrences. Random forest, inparticular, stands out as a promising algorithm for our task. It operates by constructing multiple decision trees during training and outputs the mode of the classes (stroke or no stroke) as the prediction of the individual trees. Suitable for handling complex datasets like those found in medical scenarios.

Once trained, the model will undergo rigorous evaluation using techniques like cross-validation to ensure its accuracy and generalizability. Following validation, the model will be ready for deployment in clinical settings, where it can assist health care professionals in identifying individuals at higher risk of stroke. This early identification can enable proactive intervention and lifestyle modifications to mitigate the risk of stroke and improve patient outcomes**.**

##### ADVANTAGES OF THE PROPOSED SYSTEM

The following are advantages of existing system:

* Best accuracy
* More efficiency
* Security
* No data loss
* Less time consuming (faster processing)

**FEASIBILITY STUDY**

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis:

● Economic Feasibility

● Technical Feasibility

● Social Feasibility

#### ECONOMIC FEASIBILITY

#### This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

#### TECHNICAL FEASIBILITY

#### This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

#### SOCIAL FEASIBILITY

#### The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

#### HARDWARE & SOFTWARE REQUIREMENTS

**HARDWARE REQUIREMENTS:**

Hardware interfaces specify the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

* System : I3(or)I5 version.
* Floppy Drive : 1.44 Mb.
* Monitor : 14’ClourMonitor.
* Mouse : Optical Mouse.
* Hard disk : 40GB.
* RAM : 512 Mb.

#### SOFTWARE REQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements,

* Operating system : Windows7Ultimate.
* Coding Language : Python.
* Front-End : Python.
* Designing : Html, css, javascript.
* Database : MySQL

## ARCHITECTURE

#### ARCHITECTURE

#### 

#### PROJECT ARCHITECTURE

This project architecture shows the procedure followed for classification, starting from input to final prediction.

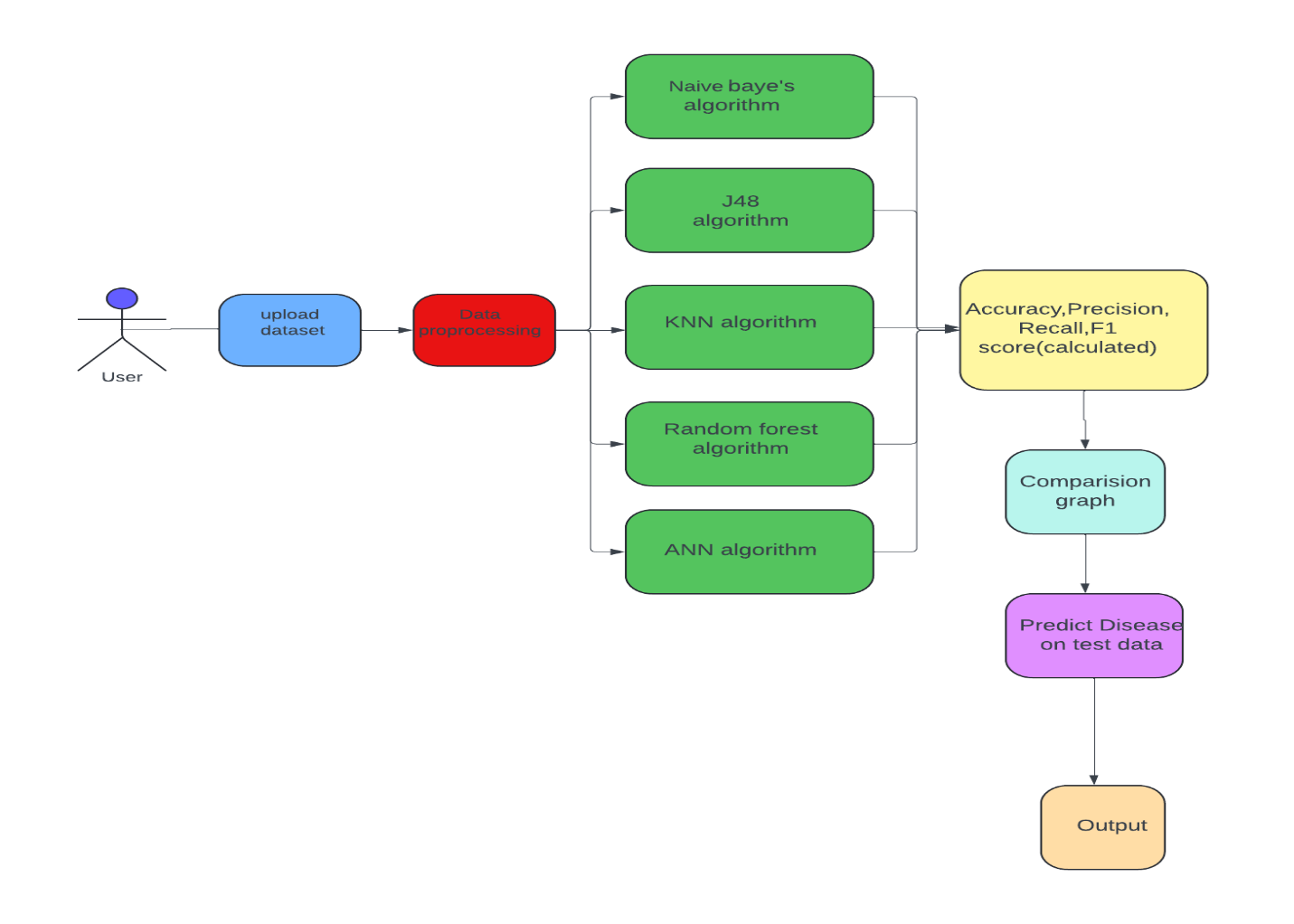


Figure3.1 : Project Architecture for Stroke risk prediction using machine learning algorithms

##### DESCRIPTION

Stroke is a significant health concern worldwide, contributing to high morbidity and mortality rates. Early identification of individuals at risk of stroke is crucial for preventive measures and timely intervention. In recent years, machine learning (ML) techniques have shown promise in predicting stroke risk by leveraging various risk factors and biomarkers

This project aims to develop a stroke risk prediction system using machine learning algorithms. By analyzing a comprehensive set of patient data, medical history, lifestyle factors the proposed system will provide personalized risk assessments for individuals. Such predictive models can aid healthcare professionals in identifying high-risk individuals, enabling targeted interventions and preventive strategies to mitigate the incidence of stroke

##### USECASEDIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. Themain purposeof ause casediagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



**system**

**user**

**importdataset**

**readdataset**

**traindataset**

**testdataset**

**predicttheresults**

**generatethegraph**

**displayresults**

Figure3.2:

Use Case Diagram for Stroke risk prediction using

machine learning algorithms

#### CLASSDIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods),and the relationships among the classes. It explains which class contains information.

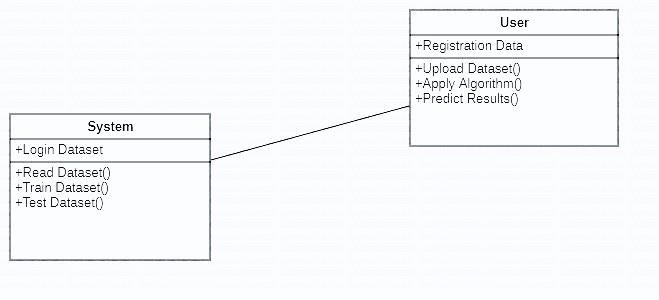


Figure3.3: Class Diagram for Stroke risk prediction using

machine learning algorithms

#### SEQUENCEDIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interactiondiagramthatshowshowprocessesoperatewithoneanotherandinwhat order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

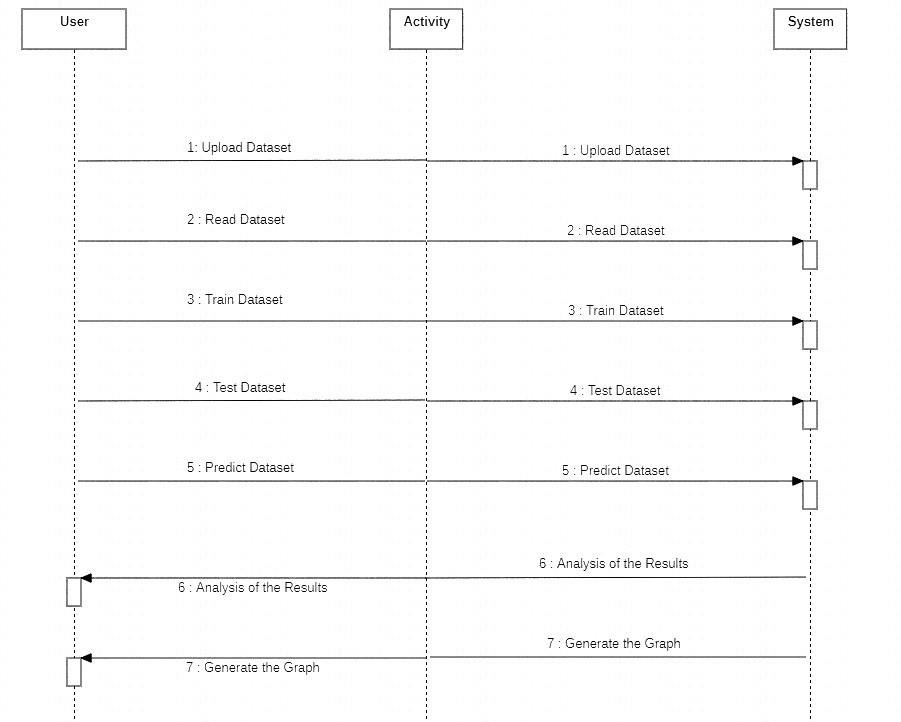


Figure3.4: Sequence diagram for Stroke risk prediction using

machine learning algorithms

##### ACTIVITYDIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



uploqaddataset

readdataset

traindataset

testdataset

predictresults

Figure3.5: Activity diagram for Stroke risk Prediction using

machine learning algorithms

# .IMPLEMENTATION

STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS

**4.IMPLEMENTATION**

#### 

#### SAMPLE CODE

From tkinter import\* import tkinter

fromtkinterimportfiledialog import numpy as np

fromtkinter.filedialogimportaskopenfilename from tkinter import simpledialog

importmatplotlib.pyplotasplt import os

importnumpyasnp importpandasaspd

from sklearn.preprocessing import LabelEncoder fromsklearn.model\_selectionimporttrain\_test\_split

fromsklearn.ensembleimportRandomForestClassifier from sklearn.metrics import accuracy\_score

fromsklearn.metricsimportprecision\_score from sklearn.metrics import recall\_score from sklearn.metrics import f1\_scoreimport seaborn as sns

from sklearn.metrics import confusion\_matrix from sklearn.naive\_bayes import GaussianNB fromsklearn.treeimportDecisionTreeClassifier

fromsklearn.neighborsimportKNeighborsClassifier from keras.utils.np\_utils import to\_categorical

fromkeras.modelsimportSequential

fromkeras.layersimportDense,Dropout,Activation

main =tkinter.Tk()

main.title("STROKERISKPREDICTIONUSINGMACHINELEARNING ALGORITHMS")

main.geometry("1000x650")

CMRTC 16

STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS

globalfilename,le1,le2,le3,le4,le5,dataset,rf global X, Y

globalX\_train,X\_test,y\_train,y\_test accuracy = []

precision=[] recall = [] fscore = []

defloadDataset():

globalfilename,dataset text.delete('1.0', END)

filename=filedialog.askopenfilename(initialdir="Dataset") text.insert(END,str(filename)+" loaded\n\n")

dataset = pd.read\_csv(filename) text.insert(END,str(dataset.head()))

def preprocessDataset(): text.delete('1.0',END) global X, Y

globalX\_train,X\_test,y\_train,y\_test global dataset, le1,le2,le3,le4,le5

le1=LabelEncoder() le2=LabelEncoder() le3=LabelEncoder() le4=LabelEncoder() le5 = LabelEncoder()

dataset.fillna(0,inplace=True)

dataset['gender']=pd.Series(le1.fit\_transform(dataset['gender'].astype(str))) dataset['ever\_married'] =

pd.Series(le2.fit\_transform(dataset['ever\_married'].astype(str)))

CMRTC 17

STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS

dataset['work\_type'] = pd.Series(le3.fit\_transform(dataset['work\_type'].astype(str)))

dataset['Residence\_type'] = pd.Series(le4.fit\_transform(dataset['Residence\_type'].astype(str)))

dataset['smoking\_status'] = pd.Series(le5.fit\_transform(dataset['smoking\_status'].astype(str)))

text.insert(END,str(dataset.head())+"\n\n") text.update\_idletasks()

label=dataset.groupby('stroke').size() dataset = dataset.values

text.insert(END,"\nTotalattributesbeforeapplyingfeaturesselection: "+str(dataset.shape[1])+"\n\n")

X=dataset[:,1:dataset.shape[1]-1] Y = dataset[:,dataset.shape[1]-1] indices = np.arange(X.shape[0]) np.random.shuffle(indices)

X=X[indices] Y = Y[indices]

text.insert(END,"\nTotalattributesafterapplyingfeaturesselection: "+str(X.shape[1])+"\n\n")

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, Y, test\_size=0.2) text.insert(END,"Total records found in dataset : "+str(X.shape[0])+"\n\n") text.insert(END,"Datasetsplitfortrainandtest.80%fortrainingand20%for

testing\n\n")

text.insert(END,"TotalrecordsusedtotrainMachineLearningAlgorithms: "+str(X\_train.shape[0])+"\n")

text.insert(END,"TotalrecordsusedtotestMachineLearningAlgorithms: "+str(X\_test.shape[0])+"\n")

label.plot(kind="bar")

plt.title("NumberofNormal&StrokeDiseaseInstancesindataset") plt.show()

CMRTC 18

STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS

defcalculateMetrics(predict,testY,algorithm):

p=precision\_score(testY,predict,average='macro')\*100 r = recall\_score(testY, predict,average='macro') \* 100

f=f1\_score(testY,predict,average='macro')\*100 a = accuracy\_score(testY,predict)\*100

text.insert(END,algorithm+' Accuracy: '+str(a)+"\n") text.insert(END,algorithm+' Precision : '+str(p)+"\n") text.insert(END,algorithm+' Recall : '+str(r)+"\n") text.insert(END,algorithm+' FScore :'+str(f)+"\n\n") accuracy.append(a)

precision.append(p) recall.append(r) fscore.append(f) text.update\_idletasks() LABELS=['Normal','Stroke']

conf\_matrix=confusion\_matrix(testY,predict) plt.figure(figsize =(6, 6))

ax=sns.heatmap(conf\_matrix,xticklabels=LABELS,yticklabels=LABELS, annot = True, cmap="viridis" ,fmt ="g");

ax.set\_ylim([0,2])

plt.title(algorithm+"Confusionmatrix") plt.ylabel('True class') plt.xlabel('Predicted class')

plt.show()

deftrainNaiveBayes():

globalX\_train,X\_test,y\_train,y\_test text.delete('1.0', END)

cls = GaussianNB() cls.fit(X\_train,y\_train)

CMRTC 19

STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS

predict = cls.predict(X\_test) calculateMetrics(predict,y\_test,"NaiveBayes")

deftrainDT():

globalX\_train,X\_test,y\_train,y\_test cls = DecisionTreeClassifier() cls.fit(X\_train, y\_train)

predict = cls.predict(X\_test) calculateMetrics(predict,y\_test,"J48Algorithm")

deftrainKNN():

globalX\_train,X\_test,y\_train,y\_test

cls=KNeighborsClassifier(n\_neighbors=2) cls.fit(X\_train, y\_train)

predict = cls.predict(X\_test) calculateMetrics(predict,y\_test,"KNN")

deftrainRanfomForest():

globalX\_train,X\_test,y\_train,y\_test,rf cls = RandomForestClassifier() cls.fit(X\_train, y\_train)

rf=cls

predict = cls.predict(X\_test) calculateMetrics(predict,y\_test,"RandomForest")

defgraph():

df=pd.DataFrame([['NaiveBayes','Precision',precision[0]],['Naive Bayes','Recall',recall[0]],['Naive Bayes','F1 Score',fscore[0]],['Naive Bayes','Accuracy',accuracy[0]],

['J48','Precision',precision[1]],['J48','Recall',recall[1]],['J48','F1 Score',fscore[1]],['J48','Accuracy',accuracy[1]],

CMRTC 20

STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS

['KNN','Precision',precision[2]],['KNN','Recall',recall[2]],['KNN','F1 Score',fscore[2]],['KNN','Accuracy',accuracy[2]],

['Random Forest','Precision',precision[3]],['Random Forest','Recall',recall[3]],['RandomForest','F1Score',fscore[3]],['Random Forest','Accuracy',accuracy[3]],

['ANN','Precision',precision[4]],['ANN','Recall',recall[4]],['ANN','F1 Score',fscore[4]],['ANN','Accuracy',accuracy[4]],

],columns=['Parameters','Algorithms','Value']) df.pivot("Parameters","Algorithms","Value").plot(kind='bar') plt.show()

deftrainANN(): global X, Y

Y1 = to\_categorical(Y)

X\_train1,X\_test1,y\_train1,y\_test1=train\_test\_split(X,Y1,test\_size=0.2) ann\_model = Sequential()

ann\_model.add(Dense(512,input\_shape=(X\_train1.shape[1],))) ann\_model.add(Activation('relu'))ann\_model.add(Dropout(0.3))

ann\_model.add(Dense(512)) ann\_model.add(Activation('relu')) ann\_model.add(Dropout(0.3)) ann\_model.add(Dense(2)) ann\_model.add(Activation('softmax'))

ann\_model.compile(loss='categorical\_crossentropy',optimizer='adam', metrics=['accuracy'])

CMRTC 21

STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS

print(ann\_model.summary())

acc\_history=ann\_model.fit(X,Y1,epochs=200, validation\_data=(X\_test1, y\_test1))

print(ann\_model.summary())

predict = ann\_model.predict(X\_test1) predict = np.argmax(predict, axis=1) testY = np.argmax(y\_test1, axis=1) calculateMetrics(predict,testY,"ANN")

def predict(): text.delete('1.0',END)

globalrf, le1,le2,le3,le4,le5

testfile=filedialog.askopenfilename(initialdir="Dataset") dataset = pd.read\_csv(testfile)

dataset.fillna(0,inplace=True)

dataset['gender']=pd.Series(le1.transform(dataset['gender'].astype(str))) dataset['ever\_married'] =

pd.Series(le2.transform(dataset['ever\_married'].astype(str))) dataset['work\_type'] =

pd.Series(le3.transform(dataset['work\_type'].astype(str))) dataset['Residence\_type'] =

pd.Series(le4.transform(dataset['Residence\_type'].astype(str))) dataset['smoking\_status'] =

pd.Series(le5.transform(dataset['smoking\_status'].astype(str))) dataset = dataset.values

dataset=dataset[:,1:dataset.shape[1]] predict = rf.predict(dataset) print(predict)

fori in range(len(predict)):

CMRTC 22

STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS

ifpredict[i] ==0:

text.insert(END,"TestData="+str(dataset[i])+"PREDICTEDAS====> NO STROKE\n\n")

ifpredict[i] ==1:

text.insert(END,"TestData="+str(dataset[i])+"PREDICTEDAS====>STROKE\n\n")

font =('times', 15, 'bold')

title=Label(main,text='STROKERISKPREDICTIONUSINGMACHINE LEARNING ALGORITHMS', justify=LEFT)

title.config(bg='lavenderblush',fg='DarkOrchid1') title.config(font=font)

title.config(height=3,width=120) title.place(x=100,y=5)title.pack()

font1 =('times', 12, 'bold')

loadButton=Button(main,text="UploadStrokeDataset", command=loadDataset)

loadButton.place(x=10,y=100) loadButton.config(font=font1)

preprocessButton=Button(main,text="DatasetPreprocessing&Features Selection", command=preprocessDataset) preprocessButton.place(x=300,y=100) preprocessButton.config(font=font1)

nbButton=Button(main,text="TrainNaiveBayesAlgorithm", command=trainNaiveBayes)

nbButton.place(x=730,y=100)

CMRTC 23

STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS

nbButton.config(font=font1)

dtButton=Button(main,text="TrainJ48Algorithm",command=trainDT) dtButton.place(x=10,y=150)

dtButton.config(font=font1)

knnButton=Button(main,text="TrainKNNAlgorithm",command=trainKNN) knnButton.place(x=300,y=150)

knnButton.config(font=font1)

rfButton=Button(main,text="TrainRandomForestAlgorithm", command=trainRanfomForest)

rfButton.place(x=730,y=150) rfButton.config(font=font1)

annButton=Button(main,text="TrainANNAlgorithm",command=trainANN) annButton.place(x=10,y=200)

annButton.config(font=font1)

graphButton=Button(main,text="ComparisonGraph",command=graph) graphButton.place(x=300,y=200)

graphButton.config(font=font1)

predictButton=Button(main,text="PredictDiseaseonTestData", command=predict)

predictButton.place(x=730,y=200) predictButton.config(font=font1)

font1 = ('times', 12, 'bold') text=Text(main,height=20,width=160) scroll=Scrollbar(text)

CMRTC 24

STROKE RISK PREDICTION USING MACHINE LEARNING ALGORITHMS

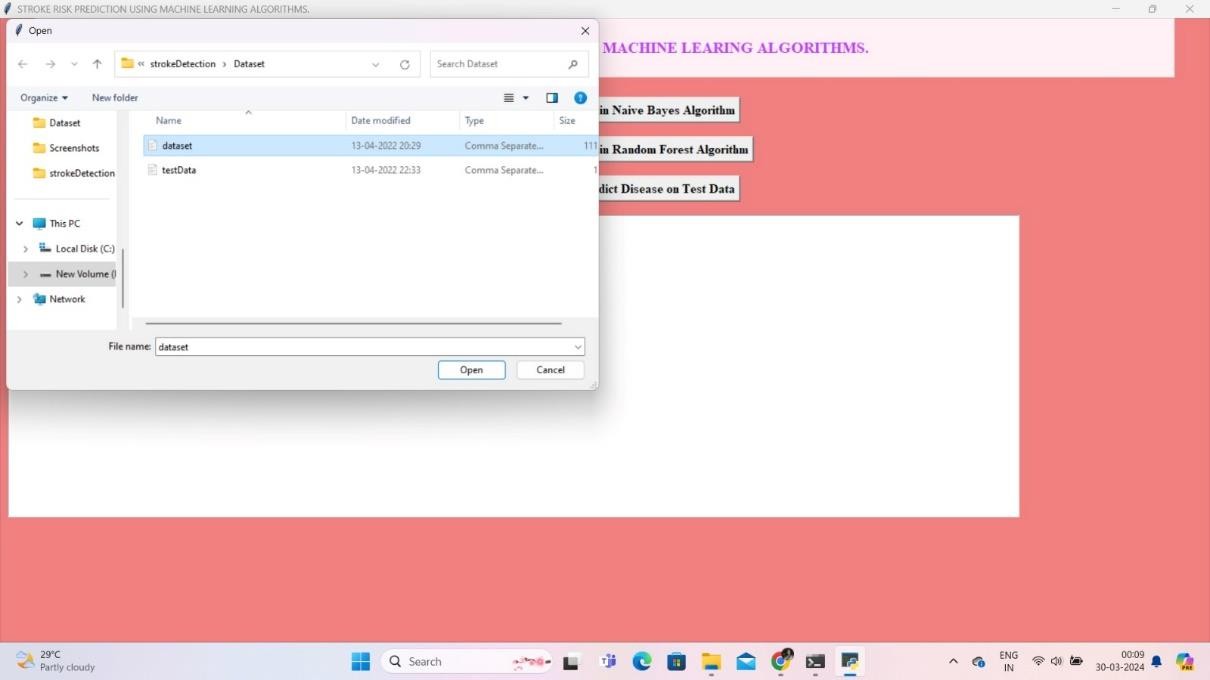
text.configure(yscrollcommand=scroll.set) text.place(x=10,y=250) text.config(font=font1)

main.config(bg='lightcoral') main.mainloop()

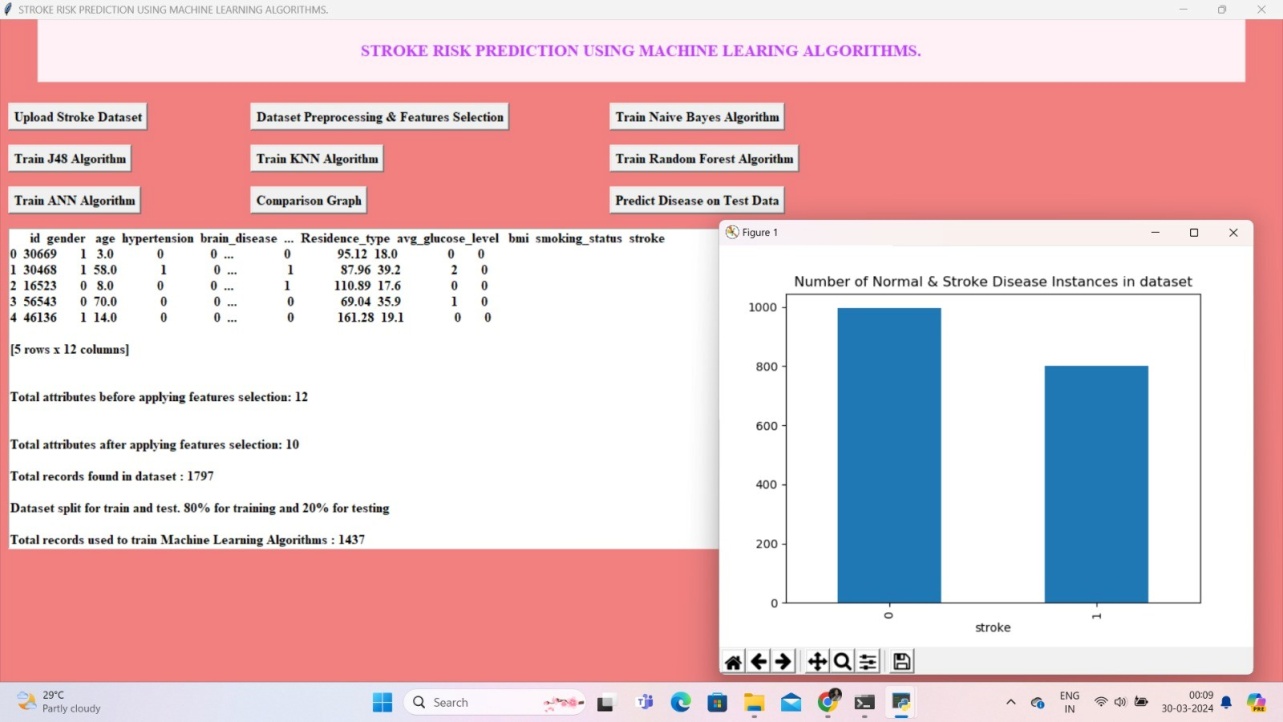
CMRTC 25

# 5.RESULTS

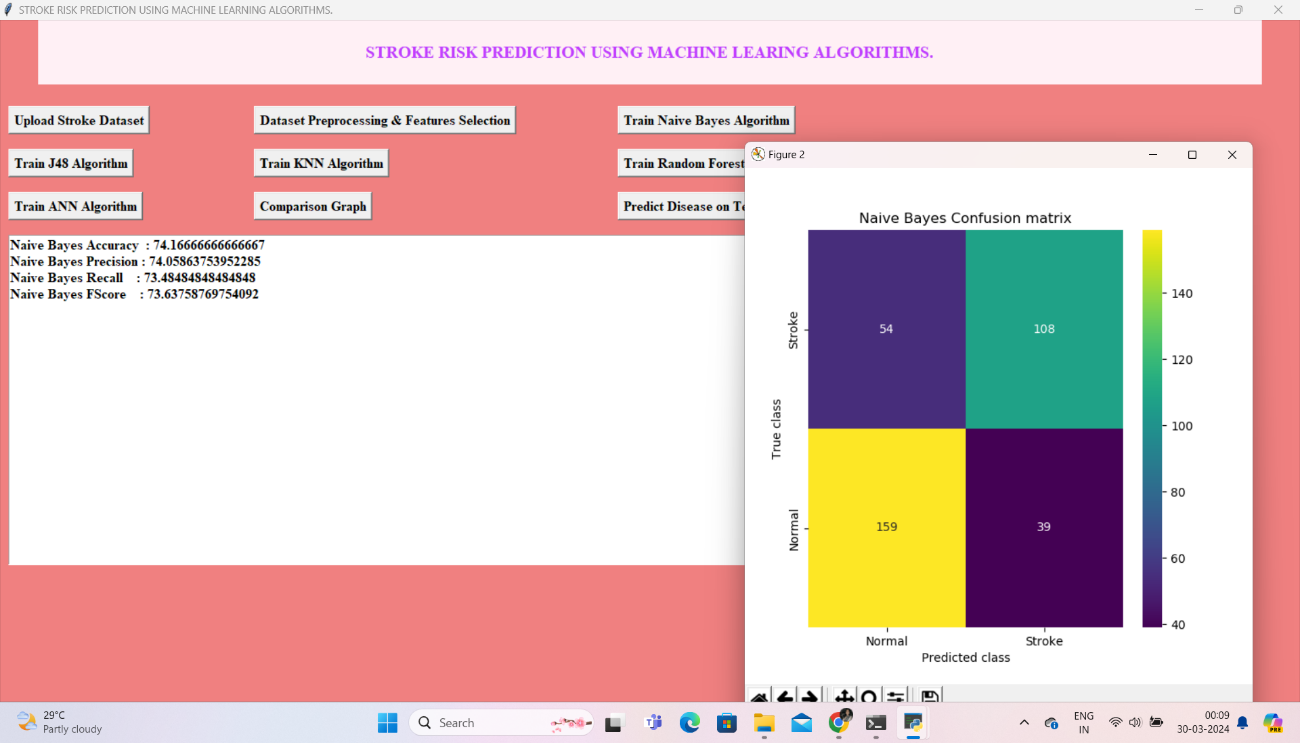
**5.RESULTS**



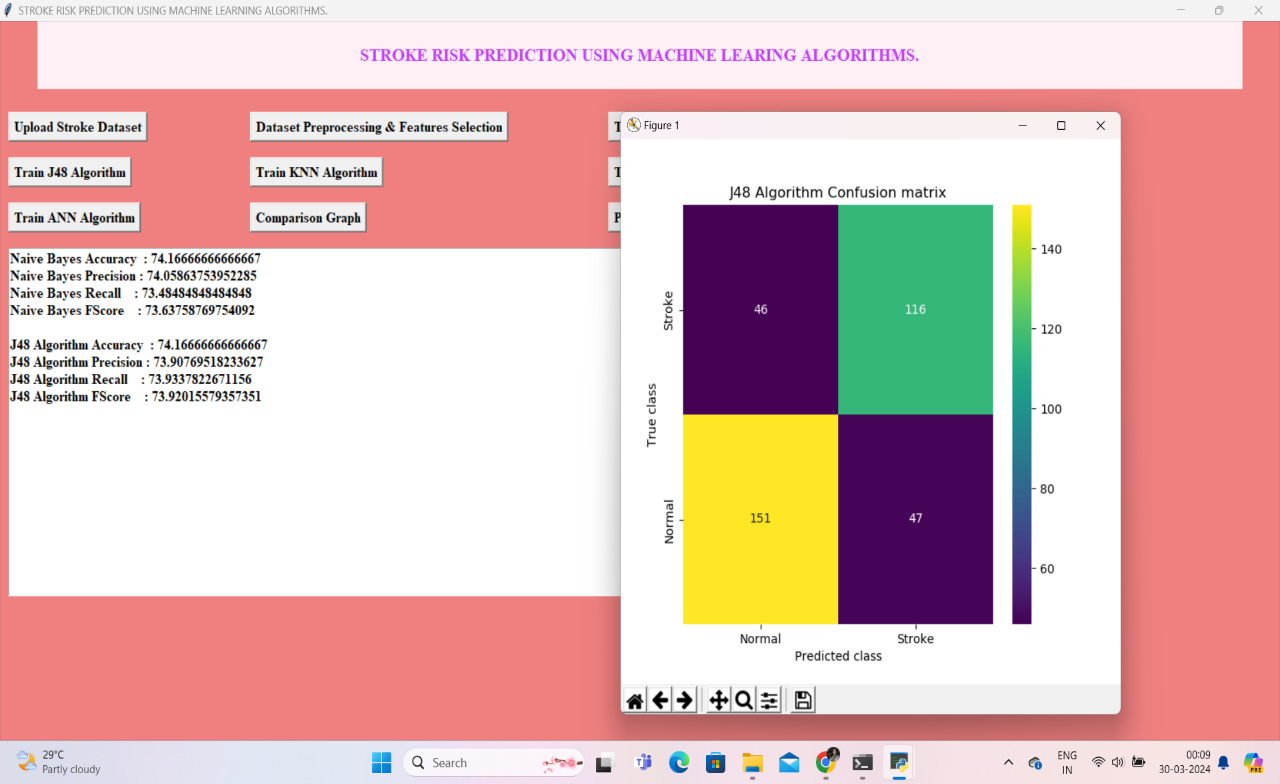
Screenshot 5.1 Upload Stroke Dataset



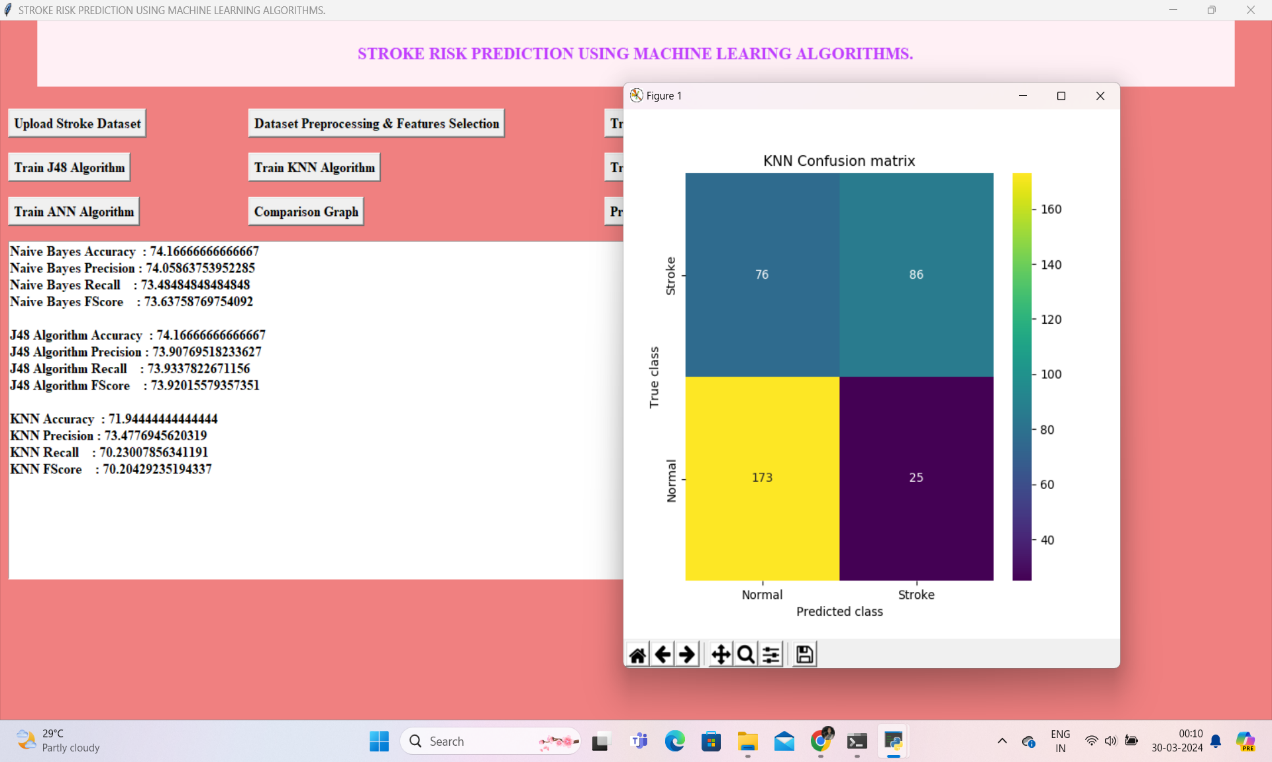
Screenshot 5.2 Dataset Preprocessing & Features Selection



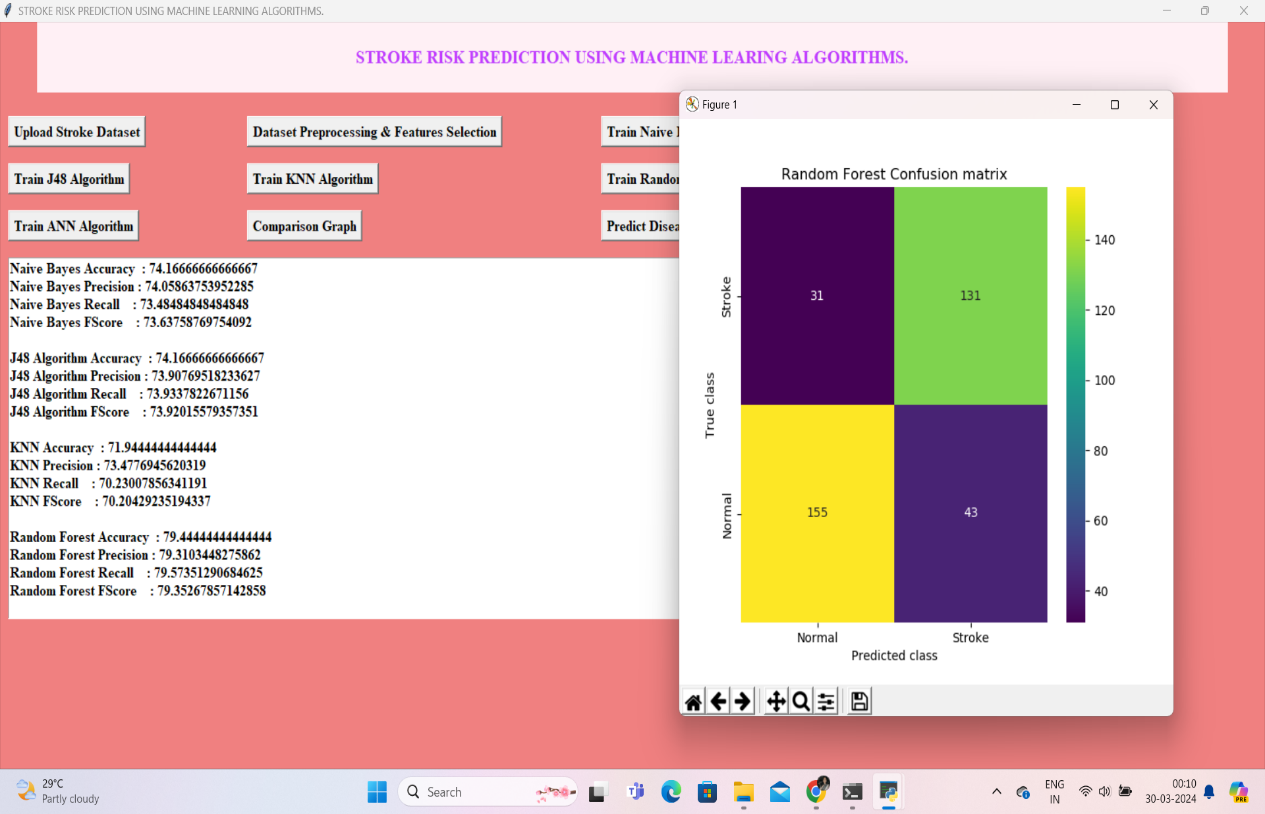
Screenshot 5.3 Train Naïve Bayes Algorithm



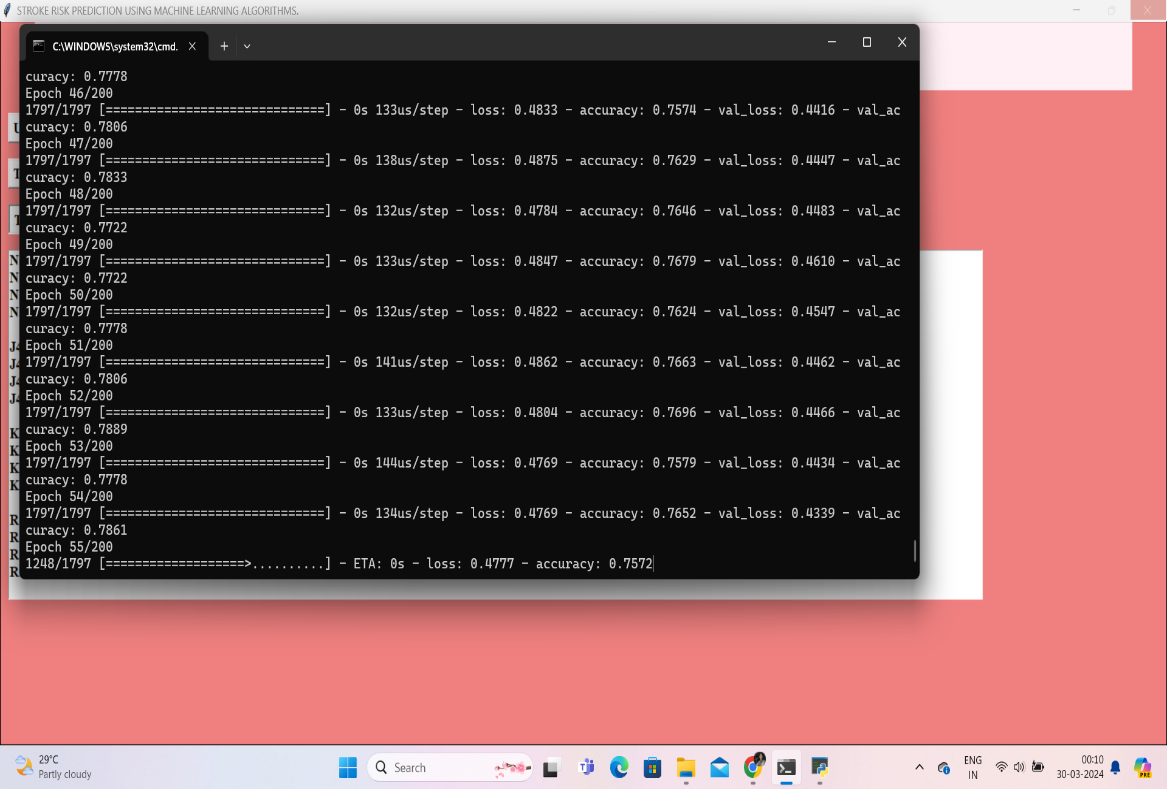
Screenshot 5.4 Train J48 Algorithm



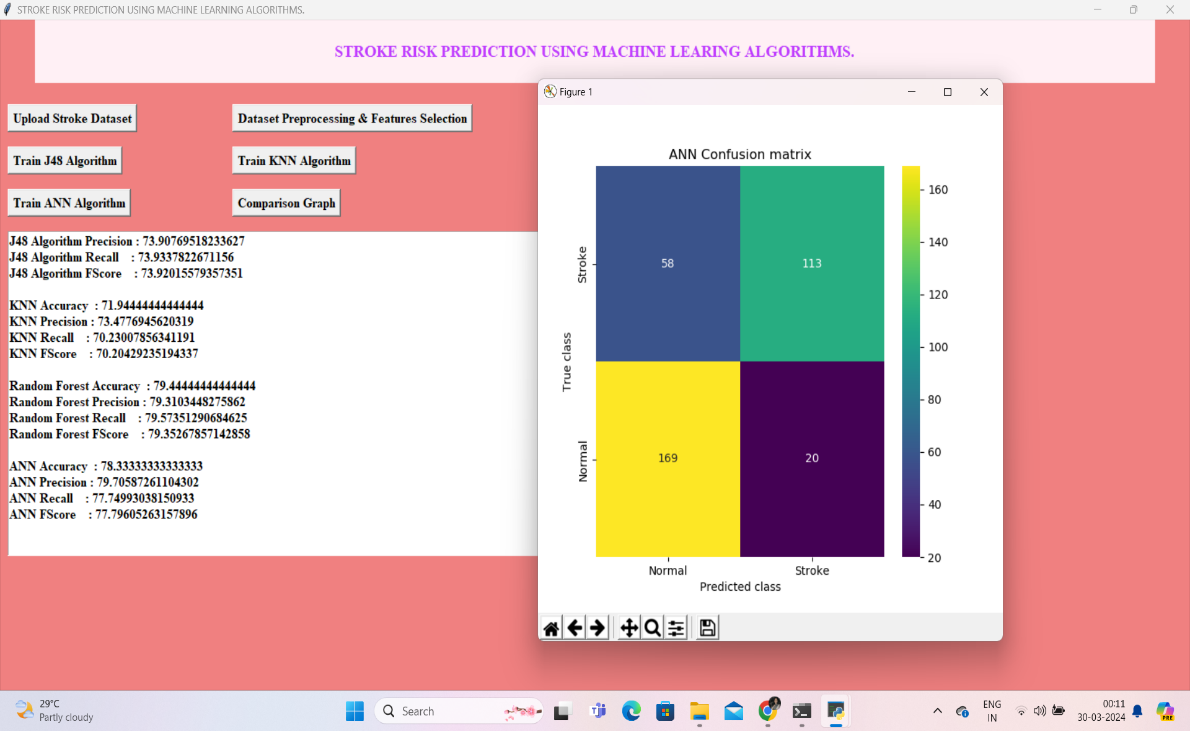
Screenshot 5.5 Run KNN Algorithm



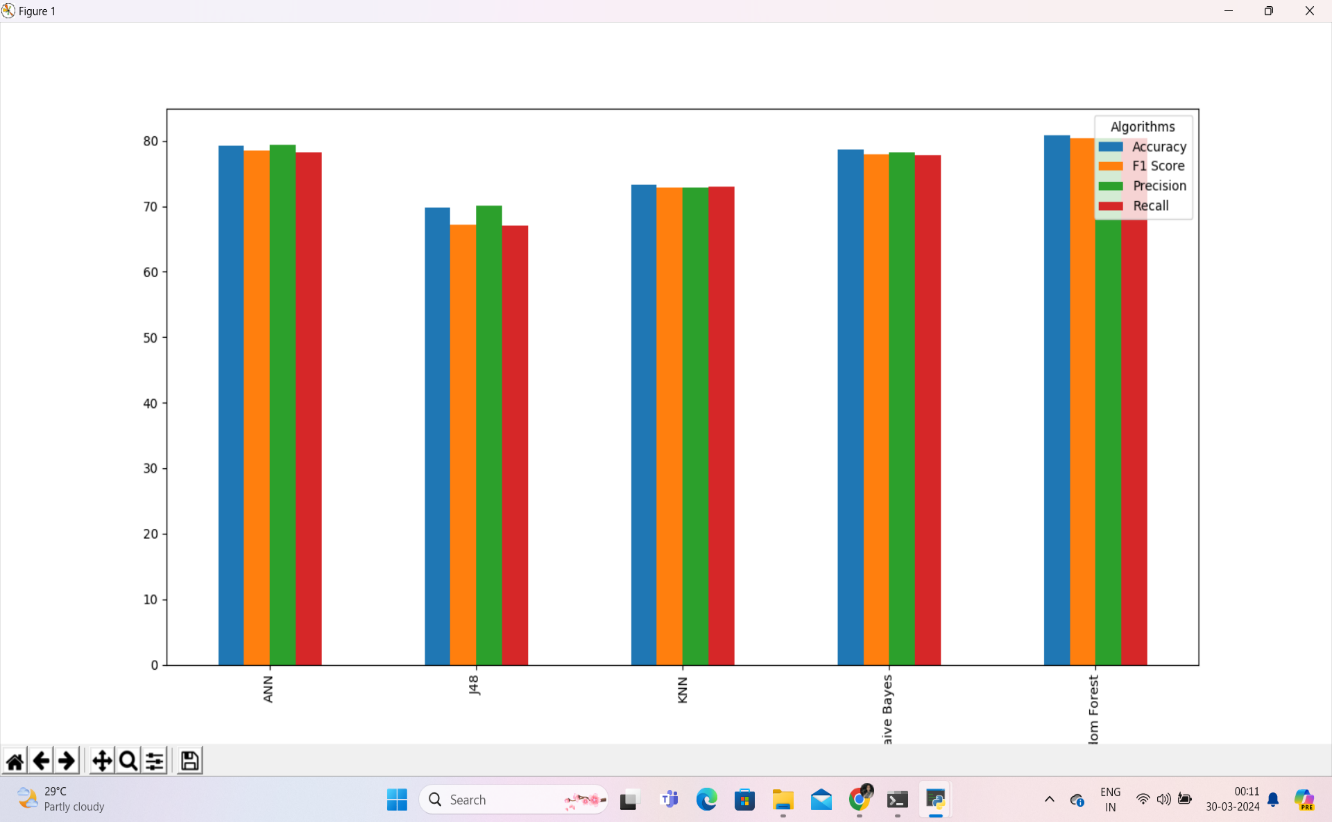
Screenshot 5.6 Run Random Forest Algorithm



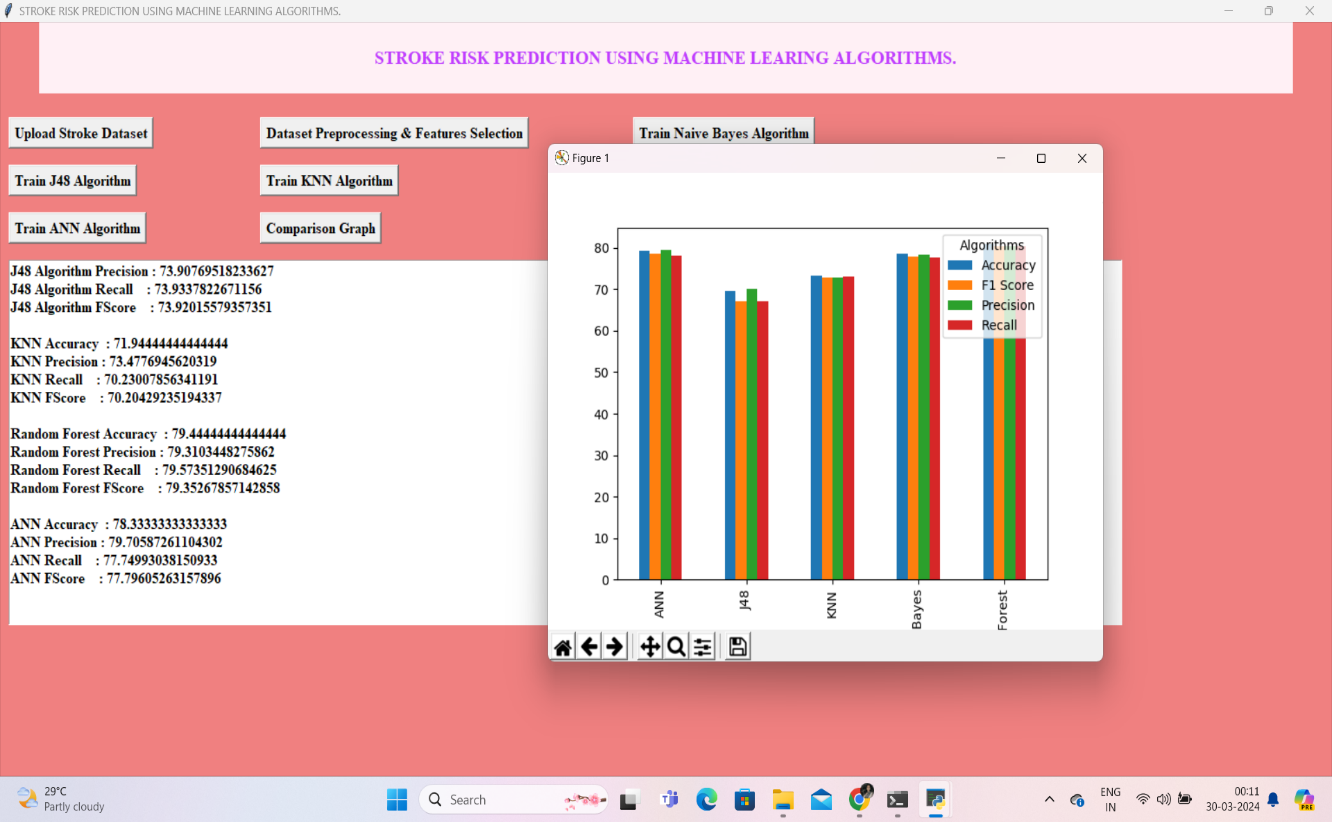
Screenshot 5.7 Command prompt



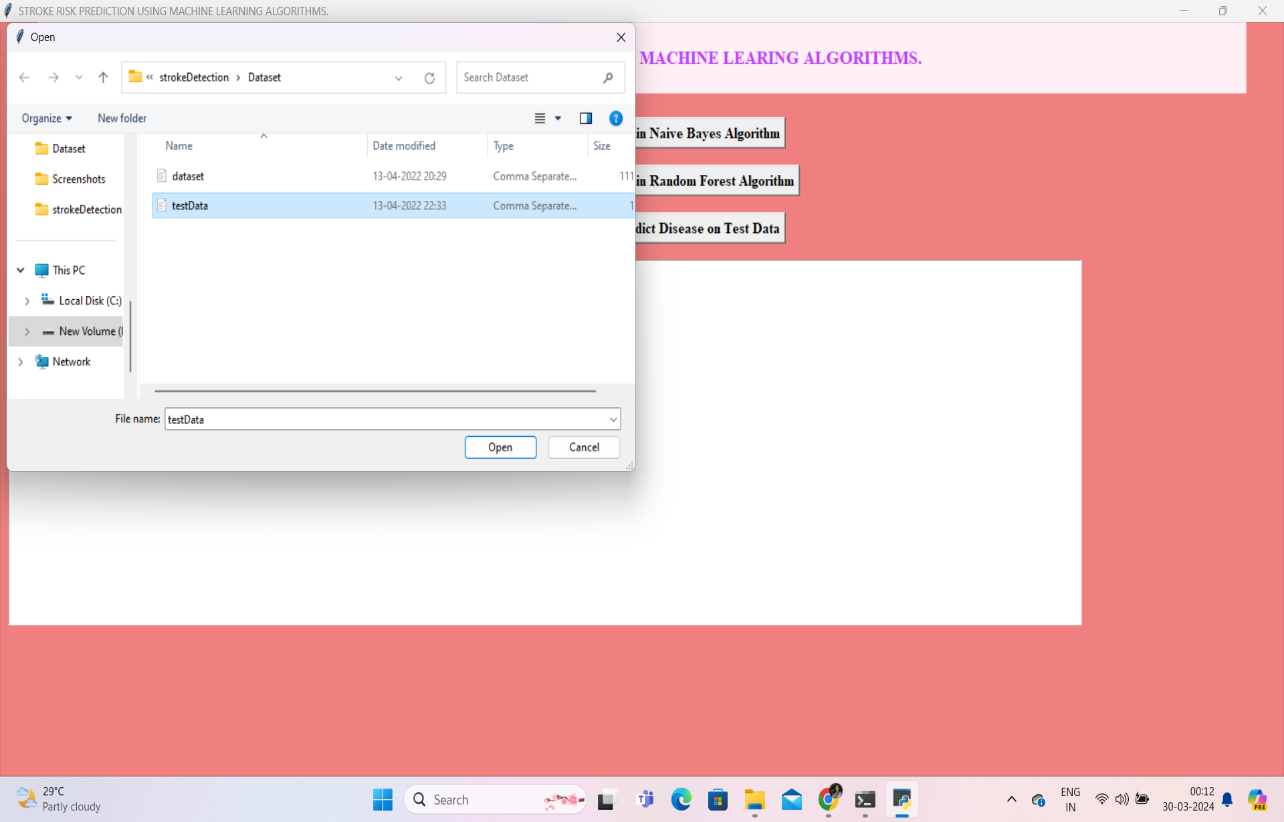
Screenshot 5.8 Run ANN Algorithm



Screenshot 5.9 Comparison Graph



Screenshot 5.10 High Accuracy Graph



Screenshot 5.11 Train Test Data



Screenshot 5.12 Prediction of Disease

**6.TESTING**

### 6.TESTING

#### INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable faultor weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not failinanun acceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

#### TYPES OF TESTS

**Unit testing:**

Unittestinginvolvesthedesignoftestcasesthatvalidatethattheinternalprogram logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

#### Integration testing:

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the

combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

###### Functional test:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted. Invalid Input :identified classes of invalid input must be rejected. Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised. Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

###### System Test:

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

###### White Box Testing:

WhiteBoxTestingisatestinginwhichinwhichthesoftwaretesterhasknowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

###### Black Box Testing:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, asmost other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

###### Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

###### Test strategy and approach

Fieldtestingwillbeperformedmanuallyandfunctionaltestswillbewrittenin

detail.

###### Test objectives

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

###### Features to be tested

* Verify that the entries are of the correct format.
* No duplicate entries should be allowed.
* All links should take the user to the correct page.

###### Integration Testing:

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

Thetaskoftheintegrationtestistocheckthatcomponentsorsoftwareapplications,

e.g. components in a software system or – one step up – software applications at the company level – interact without error.

###### Test Results:

All the testcases mentioned above passed successfully. No defects encountered.

###### Acceptance Testing:

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

###### Test Results:

All the testcases mentioned above passed successfully. No defects encountered.

**TEST CASES:**

**CLASSIFICATION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **Test Case** | **Excepted Result** | **Result** | **Remarks(IF Fails)** |
| 1. | User Register | If User registration successfully. | Pass | If already user email exist then it fails. |
| 2. | User Login | If Username and password is correct then it will getting valid  page. | Pass | Un Register Users will not logged in. |
| 3. | User View User | Show our dataset | Pass | If Dataset Not Available fail. |
| 4. | View Fast History Results | The Four Alarm  Score Should be Displayed. | Pass | The Four Alarm  Score Not Displaying fail |
| 5. | User Prediction | Display Review with true results | Pass | Results not True Fail |
| 6. | Show Detection process | Display Detection process | Pass | Results Not True Fail |
| 7. | Show Eye Blink Process | Display Eye Blink Process | Pass | If Results not Displayed Fail. |
| 8. | Admin login | Admin can login with his login credential. If success he get  His homepage | Pass | Invalid login details will not allowed here |
| 9. | Admin can  activate the register users | Admin can  activate the register user id | Pass | If user id not  Found then it won’t login |
| 10. | Results | For our Four models the accuracyandF1  Score | Pass | If Accuracy And F1 Score Not Displayed fail |

# 7.CONCLUSION

**7.CONCLUSION &FUTURESCOPE**

### CONCLUSION

In summary, the proposed Hybrid Deep Transfer Learning-based Stroke Risk Prediction (HDTL-SRP) framework tackles the challenges of data scarcity and imbalance in stroke prediction by leveraging knowledge from multiple correlated sources. Through rigorous testing, it outperforms existing models and demonstrates potential real-world deployment across multiple hospitals, aided by 5G/B5G infrastructures. The integration of Bayesian Optimization enhances parameter tuning efficiency, contributing to the robustness of the SRP model. Overall, HDTL-SRP represents a promising advancement in stroke risk prediction, addressing critical issues in healthcare data with practical implications for improved patient outcomes.

Theempiricalresultsfromextensivecomparisonswithstate-of-the-artstrokerisk prediction models in both synthetic scenarios and a real-world dataset from collaborating hospitals demonstrate the superior performance of the HDTL-SRP framework. Its ability to outperform counterparts in diverse scenarios highlights its potential for practical implementation in healthcare settings.

### FUTURESCOPE

Future research in stroke prediction using machine learning wants to make better computer programs that can tell if someone might have a stroke. They also want to make systems that can warn people early if they're at risk of having a stroke, so they can get help quickly. They're working on making these programs easy for doctors to understand and use in hospitals. Scientists also want to see if these programs really work well over time by studying them for a long time. They're also trying to make plans to help people prevent strokes based on their own personal risks, and they're making sure that everything they do follows the rules and is fair for everyone. All of this should help make strokes less common and less harmful.

# 8.BIBLIOGRAPHY

### BIBLIOGRAPHY

### REFERENCES

1. M. Mahmud and colleagues, "A brain-in spired trust management model to provide security in a cloud-based IoT framework for neuroscience applications," Cognitive Computation, vol. 10, no.5, pp. 864-873, 2018.
2. "Application of deep learning in diagnosing neurological illnesses from magnetic resonance images: a survey on the identification of Alzheimer's disease, Park in son's disease, and schizophrenia, "Brain Informatics, vol.7, no. 1, 2020, pp. 1–21.
3. A. Hussain, M. S. Kaiser, and M. Mahmud, "Deep learning in mining biological data," a rXiv preprint arXiv:2003.00108, 2020.
4. L. Amini, R. Azarpazhouh, M.T. Farzadfar, S. A.Mousavi, F. Jazaieri, F. Khorvash, R. Norouzi, and N. Toghianfar, "Prediction and control of stroke by data mining," International Journal of Preventive Medicine, vol. 4, no. Suppl 2, May 2013, pp. S245-249.
5. S. F. Sung, C Y Hsieh, Y H Kao Yang, H J Lin, and C H Chen Using data mining methods, it was possible to develop a stroke severity index based on administrative data in November 2015, according to Journal of Clinical Epidemiology, vol. 68, no. 11.
6. Low cost and portable patient monitoring system for e-health services in Bangladesh,2016 International Conference on Computer Communication and Informatics (ICCCI), pp. 1- 4. M. C. Paul, S. Sarkar, M. M. Rahman, S. M. Reza, and M. S. Kaiser.
7. "Innovative technique in web application effort and cost estimation utilising functional measurement type," in 2015 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT).IEEE (2015), pages 1–7.
8. "Toward a heterogeneous mist, fog, and cloud-based framework for the internet of healthcare things," IEEE Internet of Things Journal, vol. 6, no. 3,

pp. 4049-4062, 2018. M. Asif-Ur-Rahman, F. Afsana, M. Mahmud, M. S.Kaiser, M.

1. R. Ahmed, O.Kaiwartya, and A. James-Taylor.
2. Applications of deep learning and reinforcement learning to biological data, IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 6, pp. 2063–2079, 2018. [10] M. Mahmud, M. S. Kaiser, A. Hussain, and S. Vassanelli.
3. Tamara Islam Meghala, Md., Minazuddin Emon, and Maria Sultana Keya Performance Analysis of Machine Learning Approaches in Stroke Prediction by Mahfujur Rahman in the Fourth International Conference on Electronics, Volume8,Issue4,JulyAugust-2022|[http://ijsrcseit.com](http://ijsrcseit.com/)RishabhGurjaretal Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol., July-August-2022, 8 (4) : 2024 Communication, and Aerospace Technology (ICECA-2020).
4. "Classification of stroke disease using machine learning techniques," Neural Computing and Applications, vol.32, no.3,Feb.2020,pp.817-828.

P.Govindarajan,R.K.Soundarapandian,A.H.Gandomi,R.Patan,P.Jayaraman, and R. Manikandan.

1. In the 2014 International Conference on Electrical Engineering and Information& Communication Technology, S.M.Reza, M.M.Rahman, and

S.AlMamunpresented" A novel concept for road networks-a car xml device partnership with big data." IEEE (2014),pages 1–5.

**GITHUBLINK:**

https://github.com/Suraj5566/stroke-risk-prediction-using-machine-learning