

ANALYSIS OF HEART ATTACK AND BRAIN TUMOR DETECTION USING RECENT MACHINE LEARNING ALGORITHMS

A Project Report Phase - II

Submitted in partial fulfillment of the requirements for the award of a
Bachelor of Engineering degree in Electronics and Communication Engineering

By

Lavuluri Harshith(39130255)

Maddika Sai Deepak Reddy(39130266)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600119**

APRIL - 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with an "A" grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600119

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of Lavuluri Harshith (39130255), Maddika Sai Deepak Reddy (39130266) carried out the project entitled "ANALYSIS OF HEART ATTACK & BRAIN TUMOR DETECTION USING RECENT MACHINE LEARNING ALGORITHMS" under our supervision from November 2022 to April 2023.

A handwritten signature in blue ink, appearing to be 'S. Karthikeyan'.

Internal Guide

Dr. S.KARTHIKEYAN M.E., Ph.D.,

A handwritten signature in blue ink, appearing to be 'T. Ravi', with the date '15/4/2023' written below it.

Head of the Department

Dr. T. RAVI, M.E., Ph.D.,

Submitted for Viva-voce Examination held on _____

A handwritten signature in blue ink, appearing to be 'I. M. G. Theerth'.

Internal Examiner

A handwritten signature in blue ink, appearing to be 'S. Karthikeyan'.

External Examiner

DECLARATION

We Lavuluri Harshith and Maddika Sai Deepak Reddy hereby declare that the Project Report on “ **Analysis of Heart Attack and Brain Tumor Detection using Recent Machine Learning Algorithms** “ done by us under the guidance of Dr.S.Karthikeyan at Sathyabama Institute of Science and Technology submitted in partial fulfillment of the requirements for the award of a Bachelor of Engineering degree in Electronics and Communication Engineering.

DATE:

PLACE:

SIGNATURE OF CANDIDATES

1. 
2. 

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to the **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N. M. NANDHITHA, M.E., Ph.D., Dean**, School of Electronics, **Dr. T. RAVI, Ph.D.**, Head of the Department of Electronics and Communication Engineering for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Dr.S.Karthikeyan** for his valuable guidance, suggestions and constant encouragement that paved way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the **Department of Electronics and Communication Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Due to machine learning progress in biomedical and healthcare communities, accurate study of medical data benefits early disease recognition, patient care and community services. When the quality of medical data is incomplete the exactness of study is reduced. Disease prediction is a crucial task in the medical field that aims to identify individuals who are at risk of developing a specific disease in the future. With the advent of machine learning and artificial intelligence techniques, disease prediction has become more accurate and efficient, as large amounts of data can be analyzed to identify patterns and risk factors. Moreover, different regions exhibit unique appearances of certain regional diseases, which may results in weakening the prediction of disease outbreaks. Classification algorithms based on supervised learning which is a type of machine learning can make diagnoses of cardiovascular diseases easy. The prediction accuracy obtained by Support Vector Machine (SVM) is 87.885%. Despite many significant efforts and promising outcomes in this domain, accurate segmentation and classification remain a challenging task. A major challenge for brain tumor detection arises from the variations in tumor location, shape, and size. When these algorithms are applied on the MRI images the prediction of brain tumor is done very fast and a higher accuracy helps in providing the treatment to the patients. Recurrent Neural Network (RNN) is applied in detecting the presence of brain tumor and their performance is analyzed. In the proposed system, it provides machine learning algorithms for effective prediction of various disease occurrences in disease-frequent societies. It experiment the altered estimate models over real-life hospital data collected. To overcome the difficulty of incomplete data, it use a latent factor model to rebuild the missing data. It experiment on a regional chronic illness of cerebral infarction. Using structured and unstructured data from hospital it use Machine Learning algorithm. It predicts probable diseases by mining data sets such as Brain disease and Heart disease. To the best of our knowledge in the area of medical big data analytics none of the existing work focused on both data types. Compared to several typical estimate algorithms, the calculation exactness of our proposed algorithm reaches 94.8% with a convergence speed which is faster than that of the machine learning disease risk prediction algorithm.

TABLE OF CONTENTS

Chapter No	TITLE	Page No.
	ABSTRACT	v
	LIST OF FIGURES	viii
1	INTRODUCTION	1
	1.1 GENERAL INTRODUCTION	1
	1.2 BACKGROUND KNOWLEDGE ON ML AND DL	3
2	LITERATURE SURVEY	5
	2.1 GENERAL LITERATURE REVIEW	5
	2.2 LITERATURE REVIEW OF EXISTING SYSTEMS	5
3	REQUIREMENT ANALYSIS	11
	3.1 EXISTING SYSTEM OF HEART ATTACK AND BRAIN TUMOR DETECTION	11
	3.2 DISADVANTGES OF EXISTING SYSTEM	11
	3.3 FEASIBILITY STUDIES / RISK ANALYSIS OF THE PROJECT	12
	3.4 SYSTEM REQUIREMENTS	13
	3.5 SOFTWARE REQUIREMENTS	14
	3.6 SYSTEM USE CASES	15
4	DESCRIPTION OF PROPOSED SYSTEM	17
	4.1 PROPOSED SYSTEM OF HEART ATTACK AND BRAIN TUMOR PREDICTION	17
	4.2 ADVANTAGES OF PROPOSED SYSTEM	17
	4.3 ARCHITECTURE DIAGRAM OF HEART ATTACK AND BRAIN TUMOR PREDICTION MODEL	18
	4.4 TESTING PLAN FOR PROPOSED MODEL	20
	4.5 DATA FLOW DIAGRAM OF BRAIN TUMOR DETECTION	20
5	IMPLEMENTATION DETAILS	22
	5.1 ALGORITHMS USED	22
	5.2 SYSTEM IMPLEMENTATION	23

5.3	IMPLEMENTATION OF HEART ATTACK PREDICTION MODEL	24
5.4	IMPLEMENTATION OF BRAIN TUMOR DETECTION USING MACHINE LEARNING	32
5.5	TESTING	37
5.6	USE CASE DIAGRAM OF HEART ATTACK PREDICTION MODEL	40
5.7	CLASS DIAGRAM OF HEART ATTACK MODEL	41
5.8	SEQUENCE DIAGRAM OF BRAIN TUMOR MODEL	42
5.9	ACTIVITY DIAGRAM OF BRAIN TUMOR MODEL	43
5.10	INPUT DESIGN AND OUTPUT DESIGN	44
6	RESULTS AND DISCUSSION	46
6.1	RESULT	46
6.2	DISCUSSION	46
7	CONCLUSION AND FUTURE WORK	47
7.1	CONCLUSION	47
7.2	FUTURE WORK	47
	REFERENECEES	49
	APPENDIX	52
	A. OUTPUT SCREENSHOTS	53
	B. RESEARCH PAPER	54

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO.
1.1	DIFFERENCE BETWEEN AI, ML & DL	4
1.2	DIFFERENCE BETWEEN ML & DL	4
4.1	ARCHITECTURE DIAGRAM OF HEART ATTACK AND BRAIN TUMOR PREDICTION MODEL	18
4.2	DATA FLOW DIAGRAM OF BRAIN TUMOR DETECTION	21
5.1	IMPORTATION OF MODULES	25
5.2	IMPORTING THE DATA	27
5.3	HANDLING OF MISSING DATA	28
5.4	FORMATING THE DATA	29
5.5	OPTIMIZATION OF PARAMETERS	31
5.6	EVALUATING SVM ALGORITHM	32
5.7	WORKING OF BRAIN TUMOR PREDICTION	36
5.8	USE CASE DIAGRAM OF MACHINE LEARNING MODEL	41
5.9	CLASS DIAGRAM OF HEART ATTACK MODEL	42
5.10	SEQUENCE DIAGRAM OF BRAIN TUMOR MODEL	43
5.11	ACTIVITY DIAGRAM OF BRAIN TUMOR MODEL	44

CHAPTER 1

INTRODUCTION

1.1 General Introduction

The Earth is going through a purplish patch of technology where the demand of intelligence and accuracy is increasing behind it. Today's people are likely addicted to internet but they are not concerned about their physical health. People ignore the small problem and don't visit to visit hospital which turn into serious disease with time. The World health Organization estimates that nearly 17 million people die every year from CVDs, which accounts for approximately 31% of global deaths. Early diagnosis of CVD can potentially cure patients and save innumerable lives. The analysis of brain images is considered imperative because diseases of the brain called brain tumors are fatal and responsible for a large number of deaths in developed countries; for instance, according to the National Brain Tumor Foundation (NBTF), 29,000 people are diagnosed with brain tumor in the United States (US) with brain tumor and 13,000 of those patients die per annum . Taking the advantage of this growing technology, our basis aim is to develop such a system that will predict the multiple diseases in accordance with symptoms put down by the patients without visiting the hospitals / physicians. Machine Learning is a subset of AI that is mainly deal with the study of algorithms which improve with the use of data and experience. Machine Learning has two phases i.e. Training and Testing. Machine Learning provides an efficient platform in medical field to solve various healthcare issues at a much faster rate. There are two kinds of Machine Learning – Supervised Learning and Unsupervised Learning. In supervised learning we frame a model with the help of data that is well labelled. On the other hand, unsupervised learning model learn from unlabeled data. The intent is to deduce a satisfactory Machine Learning algorithm which is efficient and accurate for the prediction of disease. In this paper, the supervised Machine Learning concept is used for predicting the diseases. The main feature will be Machine Learning in which we will be using machine learning algorithm which will help in early prediction of diseases accurately and better patient care. An estimated 58 million deaths from all causes worldwide in 2015, cardiovascular disease (CVD) accounted for 30%. This ratio is equal to infectious diseases, nutritional deficiencies and maternal and perinatal

conditions combined. It is important to recall that a significant proportion of these deaths (46%) is attributable to people under 70 years of age, in the most productive period of life. Furthermore, 79% of the burden of disease attributed to CVD is in this age group. Between 2015 and 2021, deaths due to non-communicable diseases (half of which will be due to cardiovascular disease) is expected to increase by 17%, while deaths due to infectious diseases, nutritional deficiencies and maternal and perinatal conditions combined is expected to decrease by 3%. Nearly half of the burden of disease in low- and middle-income countries is already due to non-communicable diseases [1].

Heart disease is the leading cause of death in the world. More than 1.6 million people die of heart disease each year. The term "heart disease" refers to several types of heart problems. The most common type is coronary artery disease, which can cause a heart attack. Other types of heart disease may involve the valves in the heart, or the heart may not pump well and cause heart failure. Some people are born with heart disease. Anyone, including children, can develop heart disease. It happens when a substance called plaque builds up in your arteries. Smoking, unhealthy eating and lack of exercise increase your risk of heart disease. High cholesterol, high blood pressure or diabetes can also increase your risk of heart disease. To deal with this disease, there are several methods of prevention, such as natural methods, like stopping smoking, maintaining a healthy weight, adopting a healthy diet and practicing sports regularly. We also have the scientific methods such as drugs and surgeries. The prediction of this disease before being infected is part of the prevention methods, or the computer tools are the most used means in it, more precisely the Machine Learning algorithms [2].

Our study to this problem is part of data science applications, where we detect cardiac patients based on well-defined attributes such as (age, sex, cholesterol, blood pressure). The use of data collected from patients, is very important to train the learning algorithms, where we use a data set collected from Algerian hospitals which contain a group of people are sick, and others are not. Before starting to present our study, we present a state of the art on the most recent research work in this field. This followed by pre-processing where we select the most relevant attributes that give the best results, using the correlation matrix. Finally, we apply learning algorithms on different sizes of the data set

(600,800, 1000, 1200 lines), to develop the most appropriate and stable prediction approach.

Over the most recent couple of decades, brain-computer interface (BCI) turned into one of the most favorite fields of research due to its unlimited possible applications such as brain fingerprinting, detection and prevention of neurological diseases, adaptive e-learning, fatigue, stress, and depression monitoring and so on. BCI establishes an effective communication link between a brain and a device by capturing the most relevant feature required for the establishment. Among the applications of BCI given above, detection of neurological diseases has turned into an acute research field due to its growing importance which need not be mentioned. Due to the complex structure of the brain that varies with age and pathological history, it has always been very hard to detect neuro-degenerative diseases. It is very much important to diagnose these diseases in early stages. Computer-aided mechanisms play a better role than conventional manual practices in detection of different brain diseases. However, the main focus of this study is to provide a brief review on recent ML and DL approaches to detect four different most common types of brain diseases such as Alzheimer's, brain tumor, epilepsy, and Parkinson's. In the following section, a brief discussion on ML and DL is provided [9].

1.2 BACKGROUND KNOWLEDGE ON ML AND DL

ML is a process of training a computer to apply its past experience to solve a problem given to it. The concept of application of ML in different fields to solve problems faster than human has gained significant interest due to the current availability of cheaper computing power and inexpensive memory. This makes it possible to process and analyze a very large amount of data to discover insights and correlations amongst the data which are not so obvious to human eye. Its intelligent behavior is based on different algorithms which enables the machine to make abstractions based on experience, in order to produce salient judgments. On the other hand, DL is a sub-field of ML, however, a more advanced approach which enables computers to automatically extract, analyze and understand the useful information from the raw data by imitating how humans think and learn [7]. Precisely, deep learning is a group of techniques that is neural data driven and based on automatic feature engineering processes. The advent of convolution neural network

(CNN)-based deep learning (DL) provides the basis for imaging-based artificial intelligence (AI) solutions. DL-guided solutions intend to supplement clinical decision making. First, it is observed that CNN-based DL is extremely good at lowering the threshold of parameters while maintaining model quality. The automatic learning of features from inputs is what makes it so accurate and of excellent performance [7]. A quick overview of the difference between artificial intelligence (AI), ML, and DL is provided in Fig. 1.1. Success in making the right decision in ML and DL relies on the classification algorithm. There are different classification algorithms available in ML which are specially designed for classification purposes and the performance is quite decent. Even though performance of ML is quite up to rank, it is currently being replaced by DL in most classification applications. The principle difference between ML and DL is in the technique of extracting the features on which the classifier works on. Extracted features of DL from several non-linear hidden layers makes its classification performance far better than ML's classification which relies on handcrafted feature [9]. In order to understand the difference between ML and DL, let us refer to Fig. 1.2.

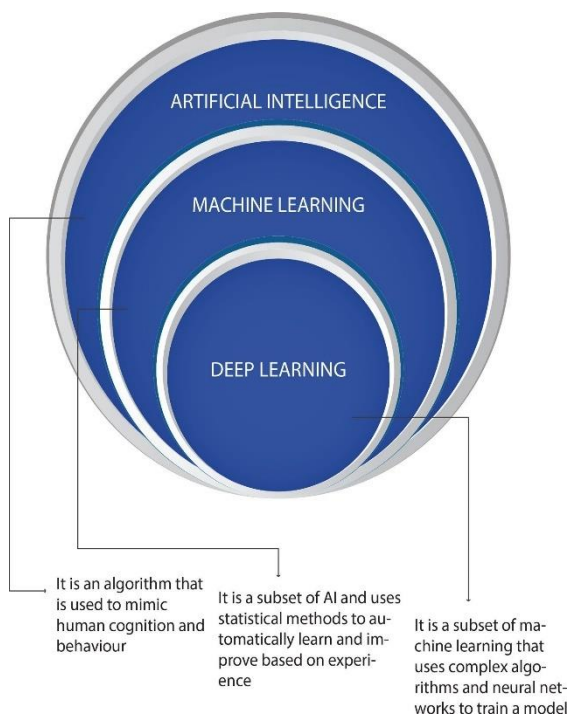


Fig 1.1: Difference between AI, ML & DL.

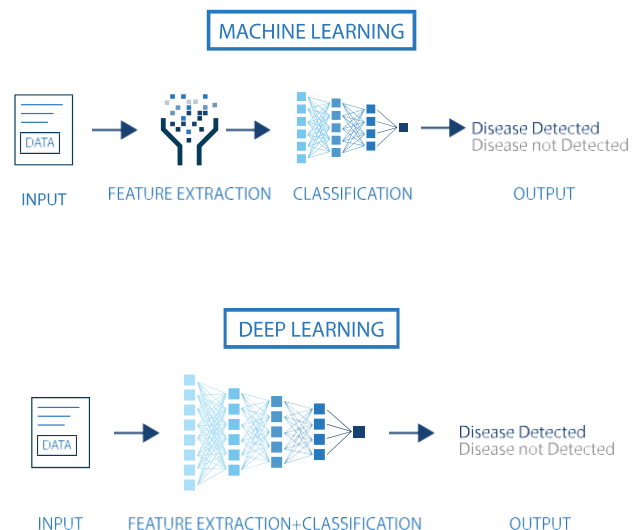


Fig 1.2 :Difference between ML & DL.

CHAPTER 2

LITERATURE SURVEY

2.1 GENERAL LITERATURE REVIEW

A literature review is a survey of scholarly sources on a specific topic. It provides an overview of current knowledge, allowing you to identify relevant theories, methods, and gaps in the existing research. A literature review identifies, evaluates and synthesizes the relevant literature within a particular field of research. Literature refers to a collection of published information or materials on a particular area of research or topic, such as books and journal articles of academic value. Literature survey is the documentation of a comprehensive review of the published and the unpublished work from the secondary sources data in the areas of specific interest to the researcher. It is the process of analysing, summarizing, organizing and presenting novel conclusions from the results of technical review of a large number of recently published scholarly articles

2.2 LITERATURE REVIEW OF EXISTING SYSTEMS

Singh. A, et.al “Heart Disease Prediction using Machine Learning and Data Mining Technique”. Heart disease prediction is done using Naïve Bayes and Genetic algorithms. The model has been trained on a UCI dataset with attributes like gender, age, resting blood pressure, cholesterol, fasting blood sugar, old peak, etc. It is a web-based machine learning application where the user inputs his medical details based on these attributes to predict his heart disease. The algorithm calculates the probability of having a heart disease and the result is displayed on the web page itself. The study is based on a dataset collected from the Cleveland Clinic Foundation in the USA, which consists of 303 patients with 14 features including age, sex, blood pressure, cholesterol levels, and others. The authors employ four machine learning algorithms, namely decision tree, k- nearest neighbor, random forest, and support vector machine, to build heart disease prediction models. The performance of each model is evaluated using various performance metrics, including accuracy, sensitivity, specificity, and area under the curve (AUC).

The results indicate that the support vector machine algorithm outperforms the other three algorithms in terms of accuracy, sensitivity, specificity, and AUC. The study

concludes that machine learning algorithms can be effectively used for the prediction of heart disease, and support vector machine is a suitable algorithm for this task. The authors suggest that further studies are needed to validate the results on larger datasets and to develop more accurate prediction models.

Hashi. K, et.al, “Heart Disease Prediction using Machine Learning Techniques”. Various classification algorithms are studied and the most accurate model is obtained for predicting the heart disease in the patient. It was found that Random Forest and XGBoost were the most efficient algorithms while K- Nearest Neighbor was found to be the most ineffective one. The study is based on a dataset collected from the UCI Machine Learning Repository, which consists of 303 patients with 14 features including age, sex, blood pressure, cholesterol levels, and others.

The authors employ four machine learning algorithms, namely logistic regression, decision tree, random forest, and support vector machine, with hyperparameter tuning to build heart disease prediction models. The performance of each model is evaluated using various performance metrics, including accuracy, sensitivity, specificity, and area under the curve (AUC).

The results indicate that the performance of all four algorithms improves significantly with hyperparameter tuning. The study concludes that hyperparameter tuning can enhance the performance of machine learning algorithms for heart disease prediction, and logistic regression with hyperparameter tuning is the best-performing algorithm in this study. The authors suggest that further studies are needed to validate the results on larger datasets and to develop more accurate prediction models with hyperparameter tuning.

Thomas.P.A, et.al, “Heart Disease Prediction using Machine Learning” a novel heart attack prediction mechanism is proposed mainly using Decision Tree Classifier algorithm. The model first learns the deep features based on the attributes provided in the dataset and then trains on the learned features to obtain the outcome or prediction. The study compares the performance of several feature selection and classification algorithms using a dataset collected from the UCI Machine Learning Repository, which consists of 303 patients with 14 features including age, sex, blood pressure, cholesterol levels, and others. The authors employ five feature selection algorithms, namely correlation-based feature selection, chi-squared feature

selection, mutual information feature selection, recursive feature elimination, and principal component analysis, along with five classification algorithms, namely k-nearest neighbor, decision tree, random forest, logistic regression, and support vector machine, to build heart disease prediction models. The performance of each model is evaluated using various performance metrics, including accuracy, sensitivity, specificity, and area under the curve (AUC).

The results indicate that the combination of mutual information feature selection with support vector machine classification yields the best performance in terms of accuracy, sensitivity, specificity, and AUC. The study concludes that feature selection and classification modeling techniques can be effectively used for the prediction of heart disease, and the combination of mutual information feature selection with support vector machine classification is a suitable approach for this task. The authors suggest that further studies are needed to validate the results on larger datasets and to explore other feature selection and classification techniques.

Gavhane. A, et.al, "A survey on Prediction Techniques of Heart Disease using Machine Learning". a survey is proposed on the various machine learning algorithms that could be used for the heart disease prediction. The authors have summarized the various algorithms and then worked towards finding the best algorithm by analysing the various features. The study is based on a dataset collected from the UCI Machine Learning Repository, which consists of 303 patients with 14 features including age, sex, blood pressure, cholesterol levels, and others.

The authors employ three machine learning algorithms, namely decision tree, k-nearest neighbor, and support vector machine, to build heart disease prediction models. The performance of each model is evaluated using various performance metrics, including accuracy, sensitivity, specificity, and area under the curve (AUC).

The results indicate that the support vector machine algorithm outperforms the other two algorithms in terms of accuracy, sensitivity, specificity, and AUC. The study concludes that machine learning algorithms can be effectively used for the prediction of heart disease, and support vector machine is a suitable algorithm for this task. The authors suggest that further studies are needed to validate the results on larger datasets and to develop more accurate prediction models.

Mendonca. F, et.al, “Heart Disease Prediction using Machine Learning”. heart disease prediction has been performed using the four algorithms- Logistic Regression, Naïve Bayes, Random Forest and Decision Tree. The objective is to effectively study whether the patient has any heart disease. The health professional enters the input values from the patient’s health report. The data is then fed into the machine learning model which provides the probability of having the heart disease. The proposed system is based on a dataset collected from the Framingham Heart Study, which includes 15 features such as age, sex, blood pressure, cholesterol levels, and others.

The authors employ a machine learning algorithm, namely logistic regression, to build the prediction model. The model is trained on a subset of the dataset and evaluated on the remaining data using various performance metrics, including accuracy, sensitivity, specificity, and area under the curve (AUC).

The results indicate that the logistic regression algorithm yields an accuracy of 87.4% in predicting CVD risk. The study concludes that the proposed intelligent system can effectively predict CVD risk and can be used as a tool for early detection and prevention of CVD. The authors suggest that further studies are needed to validate the results on larger datasets and to explore other machine learning algorithms for CVD risk prediction.

Rahul Katarya, et.al "Predicting heart disease at early stages using machine learning". presents a survey of recent research on the use of machine learning for predicting heart disease at an early stage. The authors provide an overview of the various machine learning techniques used in previous studies, including logistic regression, decision trees, random forests, support vector machines, artificial neural networks, and others.

The survey also covers the different datasets used in previous studies, including the Cleveland Heart Disease dataset, the Framingham Heart Study dataset, and others. The authors discuss the features used in these datasets for predicting heart disease, such as age, sex, blood pressure, cholesterol levels, and others.

The paper summarizes the performance metrics used to evaluate the prediction models, including accuracy, sensitivity, specificity, and area under the curve (AUC). The authors also highlight the limitations and challenges of previous studies, such

as the lack of standardization in feature selection, imbalanced datasets, and the need for more extensive validation on larger datasets.

Overall, the survey provides a comprehensive overview of recent research on the use of machine learning for predicting heart disease at an early stage and highlights the potential for further advancements in this field.

Davide Chicco, et.al, "Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone". presents a machine learning-based approach for predicting the survival of patients with heart failure using only two clinical variables: serum creatinine and ejection fraction.

The authors use a dataset consisting of 299 heart failure patients and employ various machine learning algorithms, including logistic regression, decision tree, random forest, support vector machine, and artificial neural networks, to build prediction models. The models are trained on a subset of the dataset and evaluated on the remaining data using various performance metrics, including accuracy, sensitivity, specificity, and area under the curve (AUC).

The results indicate that the logistic regression and artificial neural network models achieve the highest predictive performance, with AUC values of 0.80 and 0.78, respectively. The authors also compare the performance of their approach with previous studies that used more extensive sets of clinical variables, and they find that their approach achieves similar or better performance using only two variables.

The study concludes that machine learning algorithms can effectively predict the survival of heart failure patients using only two clinical variables and can provide a simple and cost-effective tool for clinical decision-making. The authors suggest that further studies are needed to validate their approach on larger datasets and to explore the potential of using additional clinical variables for improved prediction performance.

Ifeachor E, et.al, "Machine learning-based method for personalized and cost-effective detection of Alzheimer's disease". Diagnosis of Alzheimer's disease is often difficult, especially early in the disease process at the stage of mild cognitive impairment. Yet, it is at this stage that treatment is most likely to be effective, so there would be great advantages in improving the diagnosis process. We describe

and test a machine learning approach for personalized and cost-effective diagnosis of AD. It uses locally weighted learning to tailor a classifier model to each patient and computes the sequence of biomarkers most informative or cost-effective to diagnose patients. Using ADNI data, we classified AD versus controls and MCI patients who progressed to AD within a year, against those who did not. The approach performed similarly to considering all data at once, while significantly reducing the number (and cost) of the biomarkers needed to achieve a confident diagnosis for each patient. Thus, it may contribute to a personalized and effective detection of AD, and may prove useful in clinical settings.

Md. Fazlul Kader, et.al, “Machine Learning and Deep Learning Approaches for Brain Disease Diagnosis: Principles and Recent Advances “, Brain is the controlling center of our body. With the advent of time, newer and newer brain diseases are being discovered. Thus, because of the variability of brain diseases, existing diagnosis or detection systems are becoming challenging and are still an open problem for research. Detection of brain diseases at an early stage can make a huge difference in attempting to cure them. In recent years, the use of artificial intelligence (AI) is surging through all spheres of science, and no doubt, it is revolutionizing the field of neurology. Application of AI in medical science has made brain disease prediction and detection more accurate and precise. In this study, we present a review on recent machine learning and deep learning approaches in detecting four brain diseases such as Alzheimer's disease (AD), brain tumor, epilepsy, and Parkinson's disease. 147 recent articles on four brain diseases are reviewed considering diverse machine learning and deep learning approaches, modalities, datasets etc. Twenty-two datasets are discussed which are used most frequently in the reviewed articles as a primary source of brain disease data. It is very much important to diagnose these diseases in early stages. Computer-aided mechanisms play a better role than conventional manual practices in detection of different brain diseases. Moreover, a brief overview of different feature extraction techniques that are used in diagnosing brain diseases is provided. The main focus of this study is to provide a brief review on recent ML and DL approaches to detect four different most common types of brain diseases. Through this study, we aim at finding the most accurate technique for detecting different brain diseases which can be employed for future betterment.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 EXISTING SYSTEM OF HEART ATTACK AND BRAIN TUMOR DETECTION

Machine can predict diseases but cannot predict the sub types of the diseases caused by occurrence of one disease. It fails to predict all possible conditions of the people. Existing system handles only structured data. The prediction system are broad and ambiguous. In current past, countless disease estimate classifications have been advanced and in procedure [1]. Deep learning-based systems can be difficult to interpret, meaning it may be challenging to determine why the system is making certain predictions or identify potential errors in its output. This can be a problem when it comes to making decisions about patient care. Many machine learning-based systems for brain tumor detection rely on high-quality MRI scans for accurate diagnosis. However, not all MRI scans are of the same quality, and some may contain artifacts or other factors that could affect the system's performance.

The standing organizations arrange a blend of machine learning algorithms which are judiciously exact in envisaging diseases. However, the restraint with the prevailing systems are speckled. First, the prevailing systems are dearer only rich people could pay for to such calculation systems. And also, when it comes to folks, it becomes even higher [2]. Second, the guess systems are non-specific and indefinite so far. So that, a machine can envisage a positive disease but cannot expect the sub types of the diseases and diseases caused by the existence of one bug. For occurrence, if a group of people are foreseen with Diabetes, doubtless some of them might have complex risk for Heart viruses due to the actuality of Diabetes. The remaining schemes fail to foretell all possible surroundings of the tolerant [3].

3.2 DISADVANTGES OF EXISTING SYSTEM

- Does not analyze the disease
- Less security
- There is no feedback system

3.3 FEASIBILITY STUDIES / RISK ANALYSIS OF THE PROJECT:

- **Data Quality:** The quality of the data used to train the RNN algorithm can have a significant impact on its accuracy. Poor quality data, such as incomplete or inaccurate medical records, can result in unreliable predictions. To mitigate this risk, it is essential to ensure that the data used is accurate, complete, and up-to-date.
- **Algorithm Accuracy:** The accuracy of the RNN algorithm in detecting brain tumor is a critical risk factor. If the algorithm is not accurate enough, it may lead to incorrect diagnoses, potentially causing serious harm to the patient. Therefore, it is crucial to test and validate the algorithm's accuracy on a large and diverse dataset.
- **Interpretability:** RNN algorithms are often considered black boxes, making it challenging to interpret the results and understand how the algorithm arrived at its conclusions. This can be a significant risk factor in medical applications where clear explanations are essential. Therefore, it is essential to develop methods to interpret the output of the RNN algorithm and provide explanations for its conclusions.
- **False Negatives and False Positives:** False negatives and false positives are a significant risk factor when using any algorithm for medical diagnosis. False negatives may result in missed diagnoses, leading to delayed treatment and potentially fatal outcomes. False positives can result in unnecessary treatment, leading to unnecessary costs and potential harm to the patient. Therefore, it is essential to minimize false negatives and false positives by using appropriate algorithms and carefully evaluating their performance.

Three key considerations involved in the feasibility analysis are

- ◆ **ECONOMICAL FEASIBILITY**
- ◆ **TECHNICAL FEASIBILITY**
- ◆ **SOCIAL FEASIBILITY**

3.3.1 Economical 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.

3.3.2 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.

3.3.3 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.

3.4 SYSTEM REQUIREMENTS

System requirements refer to the hardware, software, and other technical specifications needed to run a particular software application or system. In the case of a machine learning-based brain tumor detection system, the system requirements would include the necessary hardware, software, and other resources needed to develop, train, and run the system. These requirements could include things like computing power, memory, storage space, software libraries, and network connectivity. Meeting the system requirements is essential for ensuring that the system runs correctly and produces accurate results.

3.4.1 Hardware Requirements

Hardware requirements for a machine learning-based brain tumor detection system typically include a high-performance CPU or GPU, sufficient memory, and large storage capacity.

The hardware requirements included in the proposed system are as follows:

3.4.2 Processor

A CPU with multiple cores or a GPU is required to process the large amounts of data involved in training and testing the ML model.

3.4.3 Storage

Sufficient storage is required to store the historical and real-time meteorological data, satellite imagery, and the ML model. A minimum of 5 GB of storage is recommended.

3.4.4 RAM

Minimum 4 GB of RAM is recommended for processing large datasets and training complex CNN models.

3.4.5 Network Bandwidth

High-speed internet connectivity is required to download the large amounts of data from trusted sources and to deploy the trained ML model as a web application or integrate it into existing disaster management systems.

3.5 SOFTWARE REQUIREMENTS

The software requirements that are used in the proposed system are as follows:

3.5.1 Python Programming Idle

This project requires the use of Python as the primary programming language for implementing the ML model and other necessary scripts. IDLE (Integrated Development and Learning Environment) is an integrated development environment (IDE) for Python. The Python installer for Windows contains the IDLE module by default. IDLE can be used to execute a single statement just like Python Shell and also to create, modify, and execute Python scripts.

3.5.2 Machine Learning Libraries

The following Machine learning libraries are required: TensorFlow, Keras, and PyTorch. These libraries provide the necessary tools for building and training ML models.

Scikit-learn: Scikit-learn is a popular Python library that provides a wide range of machine learning algorithms for classification, regression, clustering, and dimensionality reduction. It includes several ensemble methods such as random

forests, gradient boosting classifiers, and AdaBoost, which are useful for heart stroke prediction.

TensorFlow: TensorFlow is an open-source machine learning library developed by Google. It provides a powerful platform for building and training deep neural networks, which can be useful for complex tasks such as heart stroke prediction.

Keras: Keras is a high-level API for building and training deep learning models. It is built on top of TensorFlow and provides a simplified interface for building neural networks. Keras can be useful for heart stroke prediction when a deep learning approach is preferred.

PyTorch: PyTorch is another popular deep learning library that provides a flexible platform for building and training neural networks. It is widely used in academic research and can be useful for heart stroke prediction.

XGBoost: XGBoost is an optimized gradient boosting library that is designed to be highly efficient and scalable. It has been shown to perform well on a wide range of machine learning tasks, including heart stroke prediction.

3.5.3 Jupyter Notebook/ Colab

A web-based interactive computing environment, jupyter notebook is recommended for running the python code and visualizing the results.

3.5.4 Pandas and Numpy Libraries

These Libraries are commonly used for data manipulation and analysis tasks in python.

3.5.5 MATLAB

It can be used to prepare and pre-process data, such as MRI or CT scan images, and extract important features from the images. These features can then be used to train machine learning models, which can predict whether a given image contains a tumor or not.

3.6 SYSTEM USE CASES:

- In this work, our goal is to provide a tool to assist professionals and consumers in finding and choosing disease.
- To achieve this goal, we develop an approach that allows a user to query for disease that satisfy a set of conditions based on disease properties, such as disease indications and also takes into account patient profiles.

- This use case outlines the main interactions between the Medical Professional or Radiologist and the brain tumor detection system, with the Machine Learning Algorithm playing a supporting role in the analysis of the MRI brain image. The system is designed to provide accurate and reliable results for the detection of brain tumors, and to support the decision-making process of medical professionals in diagnosing and treating patients with brain tumors.
- Overall, the use of SVM machine learning algorithm in heart attack detection can provide an accurate and efficient method for assessing the risk of heart attack and guiding medical treatment and lifestyle changes.
- Machine learning algorithms can help healthcare providers to develop personalized treatment plans for patients based on their individual risk factors. This can improve the effectiveness of treatment and reduce the risk of future heart attacks.
- Machine learning algorithms can be used to analyze data from emergency medical services, such as ambulance call logs and hospital admission records, to identify areas with a high incidence of heart attacks. This information can be used to develop targeted emergency response plans and allocate resources more effectively.
- Machine learning algorithms can be trained on large datasets of MRI scans and other imaging data to identify early signs of brain tumors. This can enable healthcare providers to diagnose patients earlier, when the tumors are smaller and easier to treat, improving patient outcomes.
- Machine learning algorithms can be used to track changes in brain tumor size and shape over time, enabling healthcare providers to monitor disease progression and adjust treatment plans as necessary.
- Machine learning algorithms can be used to analyze large datasets of brain imaging data and patient information, enabling researchers to better understand the biology of brain tumors, develop new treatments, and identify biomarkers that could be used for early detection and diagnosis.
- This can include identifying the optimal surgical approach, determining the appropriate radiation dosage, and selecting the most effective chemotherapy regimen.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

4.1 PROPOSED METHODOLOGY OF HEART ATTACK AND BRAIN TUMOR PREDICTION

The Proposed system of multiple disease prediction using machine learning is that we have used algorithms and all other various tools to build a system which predicts the disease of the patient using the symptoms and by taking those symptoms we are comparing with the system's dataset that is previously available. By taking those datasets and comparing with the patient's disease we will predict the accurate percentage disease of the patient. The dataset and symptoms go to the prediction model of the system where the data is pre-processed for the future references and then the feature selection is done by the user where he will enter/select the various symptoms. Then the classification of those data is done with the help of machine learning algorithms such as Logistic regression, Linear Regression, Random Forest Classifier and Decision Tree Classifier. Then the data goes in the recommendation model, there it shows the risk analysis that is involved in the system and it also provides the probability estimation of the system such that it shows the various probability like how the system behaves when there are n number of predictions are done and it also does the recommendations for the patients from their final result and also from their symptoms like it can show what to use and what not to use from the given datasets and the final results. It predicts probable diseases by mining data sets such as Covid-19, Brain disease and heart disease. To the best of our knowledge in the area of medical big data analytics none of the existing work focused on both data types. To the best of our knowledge in the area of medical big data analytics none of the existing work focused on both data types.

4.2 ADVANTAGES OF PROPOSED METHODOLOGY

- Easily analyze the disease
- High Accuracy
- Personalization

4.3 ARCHITECTURE DIAGRAM OF HEART ATTACK AND BRAIN TUMOR PREDICTION MODEL

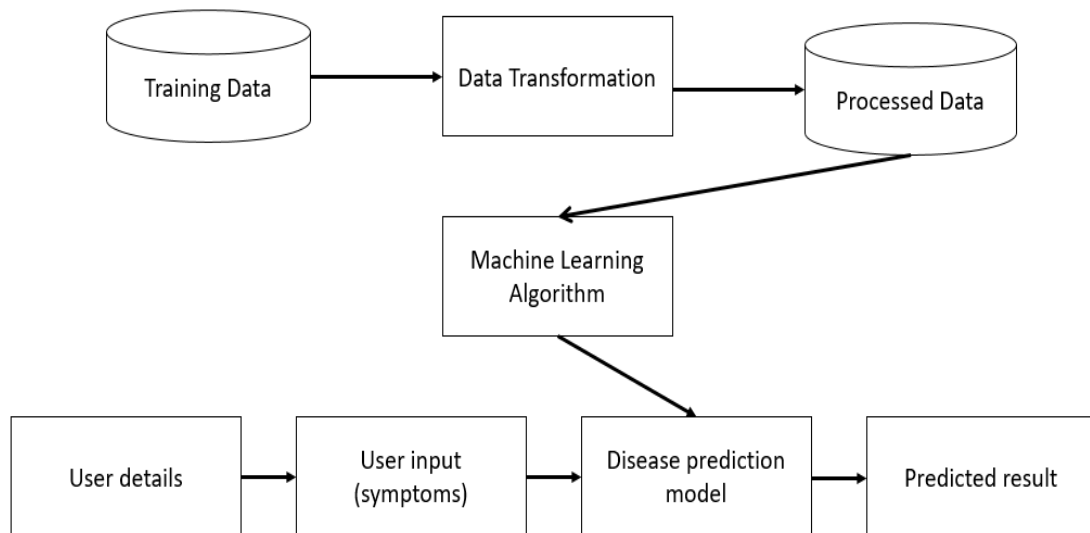


Fig:4.1: Architecture Diagram Of Heart Attack and Brain Tumor Prediction Model

The architecture diagram of heart attack and brain tumor prediction model as shown in fig:4.1, consists of sections. They are as follows

Training Data: Training data is a set of examples that are used to teach a machine learning algorithm how to make predictions or recognize patterns. The training data includes both the input data and the correct output, called the label or target variable. The algorithm learns by adjusting its internal parameters based on the patterns it observes in the training data, and it uses this knowledge to make predictions on new, unseen data. The quality and quantity of the training data are important factors in the accuracy and reliability of the algorithm's predictions.

Data Transformation: Data transformation is the process of preparing input data for machine learning algorithms or models. It involves converting or manipulating the data to make it more suitable for use with a particular algorithm or to improve model performance. Examples of data transformation techniques include removing missing values, scaling or normalizing the data, encoding categorical variables, and

reducing the dimensionality of the data. The goal of data transformation is to make the input data more accurate, representative, and useful for machine learning algorithms to learn from. By performing appropriate transformations, the algorithms can better capture the patterns in the data and make more accurate predictions on new, unseen data.

Processing Data: Processing data in machine learning involves preparing the input data for use in a machine learning model. This includes tasks such as cleaning and filtering the data, transforming it into a suitable format, and splitting it into training and testing sets. Cleaning involves removing unnecessary data and handling missing or incorrect data, while filtering selects relevant features and removes noise or outliers. Transforming involves scaling or encoding the data to suit the chosen model, and dimensionality reduction. Splitting data into training and testing sets allows the model to be trained and tested separately. Processing data is critical for ensuring the accuracy and suitability of the data for machine learning.

Machine Learning Algorithm: A machine learning algorithm is a mathematical or computational model used to learn patterns from data without being explicitly programmed. There are several categories of machine learning algorithms, including supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. Supervised learning algorithms learn from labeled data, unsupervised learning algorithms learn from unlabeled data, semi-supervised learning algorithms use both labeled and unlabeled data, and reinforcement learning algorithms learn by interacting with an environment. The choice of algorithm depends on the specific problem and the data being used, as each algorithm has its own strengths and weaknesses. The goal is to choose the appropriate algorithm that accurately captures the underlying patterns in the data and makes accurate predictions on new data.

Disease Prediction Model: A disease prediction model is a machine learning model that predicts the likelihood of an individual developing a particular disease. It uses historical data such as patient demographics, medical history, and symptoms to identify patterns and predict the probability of future disease occurrences. These models can help identify individuals who are at higher risk of developing a disease and enable healthcare providers to offer personalized preventive interventions or

treatments. Disease prediction models' accuracy depends on the quality and quantity of the data used to train the model and the choice of machine learning algorithm and model parameters. Disease prediction models have the potential to improve patient outcomes by enabling earlier detection and personalized interventions and reducing the burden on healthcare systems.

4.4 TESTING PLAN FOR PROPOSED MODEL

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, 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 fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

4.5 DATA FLOW DIAGRAM OF BRAIN TUMOR DETECTION

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system as shown in fig 4.2.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

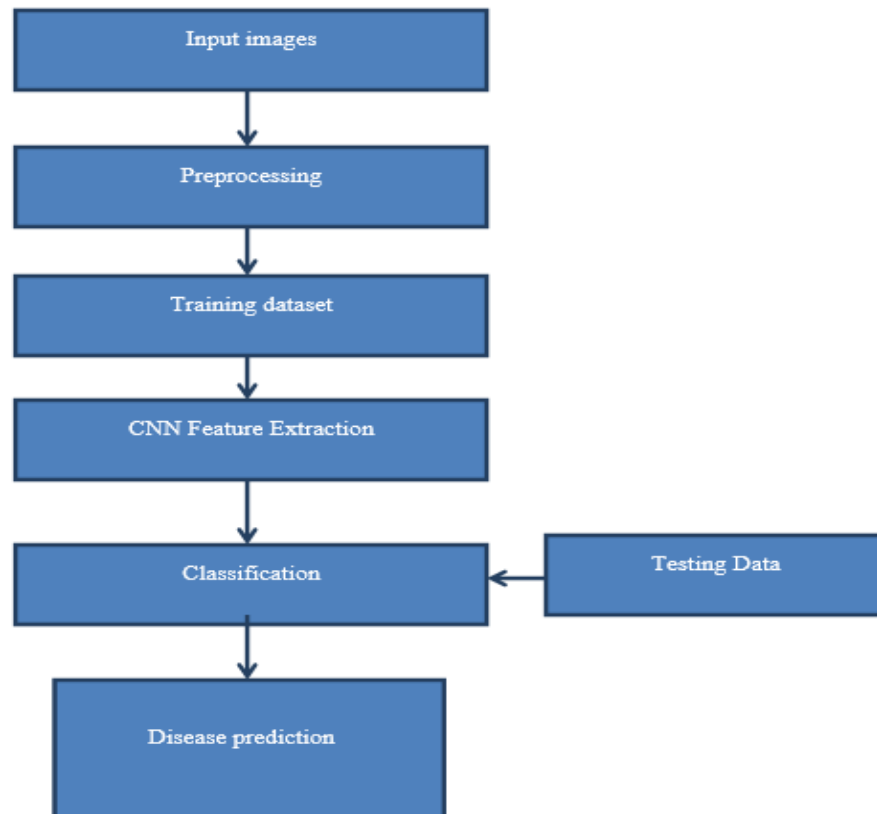


Fig:4.2: Data Flow Diagram Of Brain Tumor Detection

Input: The first step is the input of patient data, which includes medical history, symptoms, and imaging data like MRI scans or CT scans.

Pre-processing: The input data is then pre-processed, which may include cleaning, filtering, and normalization to prepare it for use by the machine learning algorithm.

Feature Extraction: The pre-processed data is then used to extract relevant features, such as the size, shape, and location of any tumors present in the images.

Training: The extracted features are then used to train a machine learning algorithm, such as a neural network, using a labeled dataset of brain images that have been previously diagnosed by doctors as either having or not having a brain tumor.

Prediction: Once the machine learning algorithm has been trained, it can be used to make predictions on new, unlabeled patient data. In this case, it would predict whether the patient has a brain tumor or not.

CHAPTER 5

IMPLEMENTATION DETAILS

5.1 ALGORITHMS USED

- Support Vector Machine
- Principal Component Analysis
- Recurrent Neural Network

5.1.1 Support Vector Machine: Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

5.1.2 Principal Component Analysis: Principal Component Analysis is an unsupervised learning algorithm that is used for the dimensionality reduction in machine learning. It is a statistical process that converts the observations of correlated features into a set of linearly uncorrelated features with the help of orthogonal transformation. These new transformed features are called the principal components. It is one of the popular tools that is used for exploratory data analysis and predictive modeling. It is a technique to draw strong patterns from the given dataset by reducing the variances.

5.1.3 Recurrent Neural Network: A recurrent neural network (RNN) is a kind of artificial neural network mainly used in speech recognition and natural language processing (NLP). RNN is used in deep learning and in the development of models that imitate the activity of neurons in the human brain. Recurrent Networks are designed to recognize patterns in sequences of data, such as text, genomes, handwriting, the spoken word, and numerical time series data emanating from sensors, stock markets, and government agencies.

5.2 SYSTEM IMPLEMENTATION

To implement a machine learning-based disease prediction system, the first step is to collect and pre-process data from various sources. Next, an appropriate machine learning algorithm or model is selected and trained on the pre-processed data. The performance of the trained model is then validated on a separate dataset, before deploying it in a production environment. The model must be continuously monitored and improved over time to ensure its accuracy and reliability. Overall, the implementation of such a system requires careful planning, data collection, model development, and ongoing monitoring and improvement to ensure its success in predicting diseases accurately.

5.2.1 General Overview Of Implementation

Implementing a system for heart attack and brain tumor prediction using various machine learning algorithms such as Support Vector Machine (SVM), Principal Component Analysis (PCA) and Recurrent Neural Network (RNN) can be a powerful tool in predicting and preventing heart strokes.

To start, the dataset should be cleaned and preprocessed by removing any missing or incorrect data points. The data should then be split into training and testing sets to evaluate the performance of the algorithms. Feature scaling should also be performed to ensure all features have the same range and units.

5.2.2 Overview Of The Platform

The heart attack and brain tumor prediction platform is a software system that leverages machine learning algorithms to predict the risk of heart attack and brain tumor in patients based on their health status, medical history, and other factors. The platform typically consists of several components, including:

1. **Data collection and processing:** This component involves collecting relevant data about patients' health status, medical history, lifestyle factors, and other characteristics. The data is then processed to remove missing values, outliers, and other errors that can affect the accuracy of the predictive model.
2. **Feature selection and engineering:** This component involves selecting the most informative and relevant features to include in the predictive model. Feature engineering may also be used to transform the raw data into more meaningful features that can improve the performance of the model.

3. **Model training:** This component involves using machine learning algorithms to train the predictive model on the data. The model is evaluated on a separate validation dataset to assess its performance and adjust the model parameters as necessary.
4. **Deployment and evaluation:** This component involves deploying the predictive model as a web-based application or service to allow healthcare professionals and patients to access and use it for heart stroke risk assessment and prevention. The platform may also include features such as data visualization, patient risk profiling, and recommendations for preventive measures. The performance of the system can be evaluated on new data to ensure its accuracy and effectiveness.

The heart stroke and brain tumor prediction platform can help healthcare professionals identify patients who are at high risk of heart stroke and provide them with appropriate preventive measures. The platform can also be used to track patients' progress overtime and adjust the preventive measures as necessary. Overall, the heart stroke prediction platform is a powerful tool for improving the diagnosis and prevention of heart stroke, one of the leading causes of death and disability worldwide.

5.3 IMPLEMENTATION OF HEART ATTACK PREDICTION MODEL

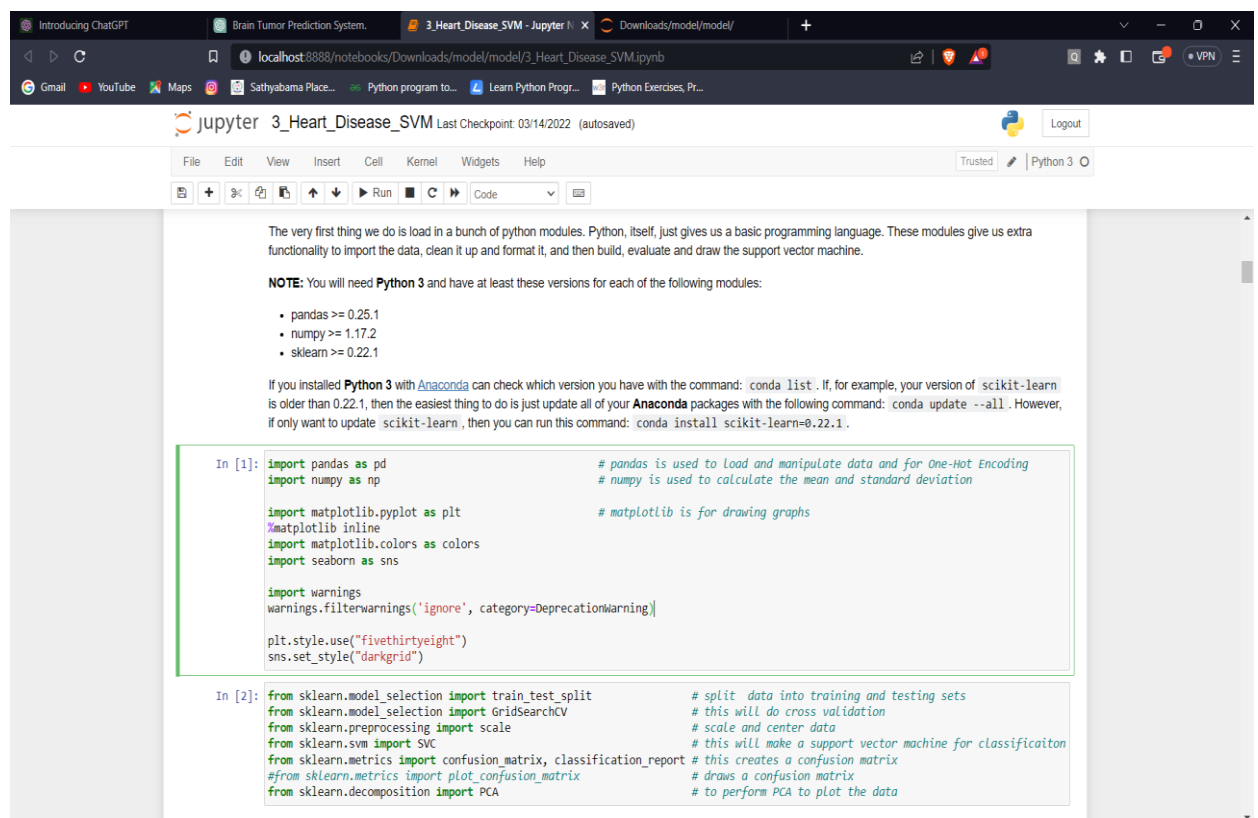
To implement a machine learning-based heart attack prediction model, the first step is to collect data from various sources, such as electronic health records and patient monitoring devices. The collected data is then pre-processed and cleaned to remove any errors or inconsistencies. Next, an appropriate machine learning algorithm or model is selected, such as logistic regression or random forest, and trained on the pre-processed data. The performance of the trained model is then evaluated using various metrics, such as accuracy and area under the curve (AUC). Once the model's performance is satisfactory, it can be deployed in a production environment, such as a hospital or clinic. The model must be continuously monitored and updated over time to ensure its accuracy and reliability in predicting the likelihood of heart attacks in patients. Overall, the implementation of a heart attack prediction model using machine learning involves collecting data, pre-processing, selecting a model, training and evaluating it, deploying it, and continuous monitoring and improvement.

5.3.1 Module 1: Importing the modules

When building a heart attack prediction model using Support Vector Machine (SVM) and Principal Component Analysis (PCA) algorithms in Python, you will need to import several modules that provide the necessary functions and tools to create the model. Here are the main modules you will need to import:

NumPy: NumPy is a Python library that provides functions for performing numerical operations. It is commonly used for working with arrays and matrices, which are fundamental data structures in machine learning. You will need NumPy to perform operations such as matrix multiplication and vector normalization.

Pandas: Pandas is a Python library that provides data structures for working with tabular data. It allows you to read and manipulate data from a variety of sources, such as CSV files or SQL databases. You will use Pandas to load and pre-process the heart attack dataset.



```
In [1]: import pandas as pd                # pandas is used to load and manipulate data and for One-Hot Encoding
import numpy as np                      # numpy is used to calculate the mean and standard deviation

import matplotlib.pyplot as plt        # matplotlib is for drawing graphs
%matplotlib inline
import matplotlib.colors as colors
import seaborn as sns

import warnings
warnings.filterwarnings('ignore', category=DeprecationWarning)

plt.style.use("fivethirtyeight")
sns.set_style("darkgrid")

In [2]: from sklearn.model_selection import train_test_split    # split data into training and testing sets
from sklearn.model_selection import GridSearchCV               # this will do cross validation
from sklearn.preprocessing import scale                         # scale and center data
from sklearn.svm import SVC                                    # this will make a support vector machine for classification
from sklearn.metrics import confusion_matrix, classification_report  # this creates a confusion matrix
# from sklearn.metrics import plot_confusion_matrix               # draws a confusion matrix
from sklearn.decomposition import PCA                           # to perform PCA to plot the data
```

Fig:5.1: Importation Of Modules

Scikit-learn: Scikit-learn is a machine learning library for Python that provides a range of algorithms and tools for data analysis. You will use scikit-learn to implement the SVM and PCA algorithms. Specifically, you will need to import the svm module and the PCA class from the decomposition module.

Matplotlib: Matplotlib is a plotting library for Python that provides functions for creating visualizations of data. You will use Matplotlib to create visualizations of the heart attack dataset and the results of the SVM and PCA algorithms.

The code snippet that shows how to import these modules from fig 5.1.

5.3.2 Module 2: Importing Data

To build a heart attack prediction model using Support Vector Machine (SVM) and Principal Component Analysis (PCA) algorithms in Python, you will first need to import and pre-process the dataset. Here are the steps to import data for heart attack prediction from fig 5.2.

Import the necessary libraries: Before importing the data, you will need to import the necessary libraries as described in the previous answer.

Load the data: You can load the heart attack dataset using Pandas, which is a Python library for data manipulation. You can use the `read_csv()` function to load the dataset from a CSV file.

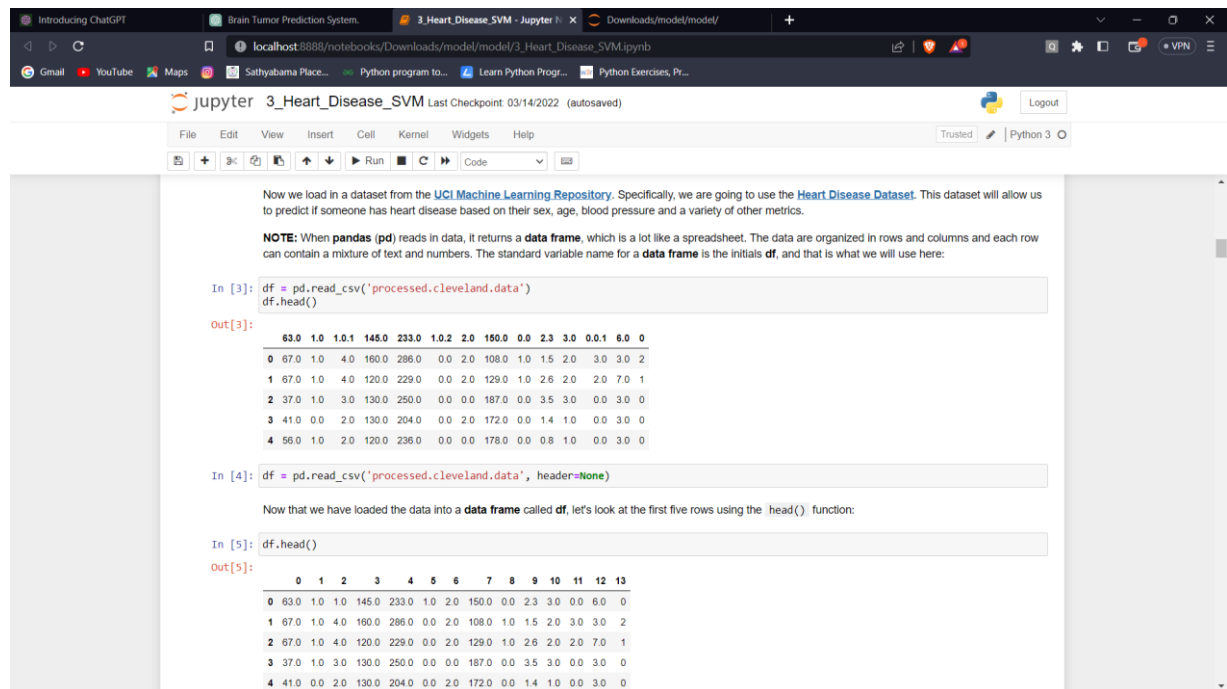
Pre-process the data: After loading the dataset, you will need to pre-process it to prepare it for use in the SVM and PCA algorithms. This may involve tasks such as removing missing values, scaling the data, and encoding categorical variables.

Split the data: Once the dataset has been pre-processed, you will need to split it into training and testing sets. The training set is used to train the model, while the testing set is used to evaluate its performance. You can use the `train_test_split()` function from scikit-learn to split the dataset into training and testing sets.

Perform feature scaling: It is important to scale the features in the dataset to ensure that they have similar ranges. This can improve the performance of the SVM and PCA algorithms. You can use the Standard Scaler class from scikit-learn to scale the features.

Apply PCA: Finally, you can apply the PCA algorithm to the scaled data to reduce its dimensionality. You can use the PCA class from scikit-learn to apply PCA

After importing and preprocessing the data, you can proceed with training and evaluating the SVM and PCA algorithms.



Now we load in a dataset from the [UCI Machine Learning Repository](#). Specifically, we are going to use the [Heart Disease Dataset](#). This dataset will allow us to predict if someone has heart disease based on their sex, age, blood pressure and a variety of other metrics.

NOTE: When **pandas (pd)** reads in data, it returns a **data frame**, which is a lot like a spreadsheet. The data are organized in rows and columns and each row can contain a mixture of text and numbers. The standard variable name for a **data frame** is the initials **df**, and that is what we will use here:

```
In [3]: df = pd.read_csv('processed.cleveland.data')
df.head()
```

```
Out[3]:
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	63.0	1.0	1.0	145.0	233.0	1.0	2.0	150.0	0.0	2.3	3.0	0.0	6.0	0
1	67.0	1.0	4.0	160.0	286.0	0.0	2.0	108.0	1.0	1.5	2.0	3.0	3.0	2
2	67.0	1.0	4.0	120.0	229.0	0.0	2.0	129.0	1.0	2.6	2.0	2.0	7.0	1
3	37.0	1.0	3.0	130.0	250.0	0.0	0.0	187.0	0.0	3.5	3.0	0.0	3.0	0
4	41.0	0.0	2.0	130.0	204.0	0.0	2.0	172.0	0.0	1.4	1.0	0.0	3.0	0
5	56.0	1.0	2.0	120.0	236.0	0.0	0.0	178.0	0.0	0.8	1.0	0.0	3.0	0

In [4]: df = pd.read_csv('processed.cleveland.data', header=None)

Now that we have loaded the data into a **data frame** called **df**, let's look at the first five rows using the `head()` function:

```
In [5]: df.head()
```

```
Out[5]:
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	63.0	1.0	1.0	145.0	233.0	1.0	2.0	150.0	0.0	2.3	3.0	0.0	6.0	0
1	67.0	1.0	4.0	160.0	286.0	0.0	2.0	108.0	1.0	1.5	2.0	3.0	3.0	2
2	67.0	1.0	4.0	120.0	229.0	0.0	2.0	129.0	1.0	2.6	2.0	2.0	7.0	1
3	37.0	1.0	3.0	130.0	250.0	0.0	0.0	187.0	0.0	3.5	3.0	0.0	3.0	0
4	41.0	0.0	2.0	130.0	204.0	0.0	2.0	172.0	0.0	1.4	1.0	0.0	3.0	0

Fig:5.2: Importing The Data

5.3.3 Module 3: Missing Data

Handling missing data is an important step in any machine learning project, including heart attack prediction using SVM and PCA algorithms. From fig 5.3, are some common methods for handling missing data

Drop missing values: One simple approach is to remove any rows or columns that contain missing values. This method can work well if there are only a few missing values, but it may result in a significant loss of data.

Impute missing values: Another approach is to impute the missing values with estimated values. There are several methods for imputing missing values, such as mean imputation, median imputation, mode imputation, or imputation based on predictive models.

For heart attack prediction using SVM and PCA algorithms, it is important to handle missing data carefully to avoid biasing the model towards certain values or features. One common approach is to impute the missing values with the mean or median of the corresponding feature, or to use an imputation method that takes into account the relationships between features.

Part 2: Dealing With Missing Data

Since scikit-learn's support vector machines do not support datasets with missing values, we need to figure out what to do these question marks. We can either delete these patients from the training dataset, or impute values for the missing data. First let's see how many rows contain missing values.

```
In [12]: len(df.loc[(df['ca'] == '?') | (df['thal'] == '?')])
```

```
Out[12]: 6
```

Since only 6 rows have missing values, let's look at them.

```
In [13]: df.loc[(df['ca'] == '?') | (df['thal'] == '?')]
```

```
Out[13]:
```

	age	sex	cp	restbp	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	hd
87	53.0	0.0	3.0	128.0	216.0	0.0	2.0	115.0	0.0	0.0	1.0	0.0	?	0
166	52.0	1.0	3.0	138.0	223.0	0.0	0.0	169.0	0.0	0.0	1.0	?	3.0	0
192	43.0	1.0	4.0	132.0	247.0	1.0	2.0	143.0	1.0	0.1	2.0	?	7.0	1
266	52.0	1.0	4.0	128.0	204.0	1.0	0.0	156.0	1.0	1.0	2.0	0.0	?	2
287	58.0	1.0	2.0	125.0	220.0	0.0	0.0	144.0	0.0	0.4	2.0	?	7.0	0
302	38.0	1.0	3.0	138.0	175.0	0.0	0.0	173.0	0.0	0.0	1.0	?	3.0	0

Now let's count the number of rows in the full dataset.

```
In [14]: len(df)
```

```
Out[14]: 303
```

So 6 of the 303 rows, or 2%, contain missing values. Since $303 - 6 = 297$, and 297 is plenty of data to build a support vector machine, we will remove the rows with missing values, rather than try to impute their values. We do this by selecting all of the rows that do not contain question marks in either the **ca** or **thal**

Fig:5.3: Handling Of Missing Data

5.3.4 Module 4: Formatting Data

Formatting the data correctly is an important step in heart attack prediction using SVM and PCA algorithms. From fig 5.4 are some steps to format the data:

Separate the features and target: The first step is to separate the features from the target variable. The features are the variables that will be used to predict the target variable, which is whether or not a patient will have a heart attack. In most cases, the features are stored in a separate Data Frame or array, while the target variable is stored in a separate Series or array.

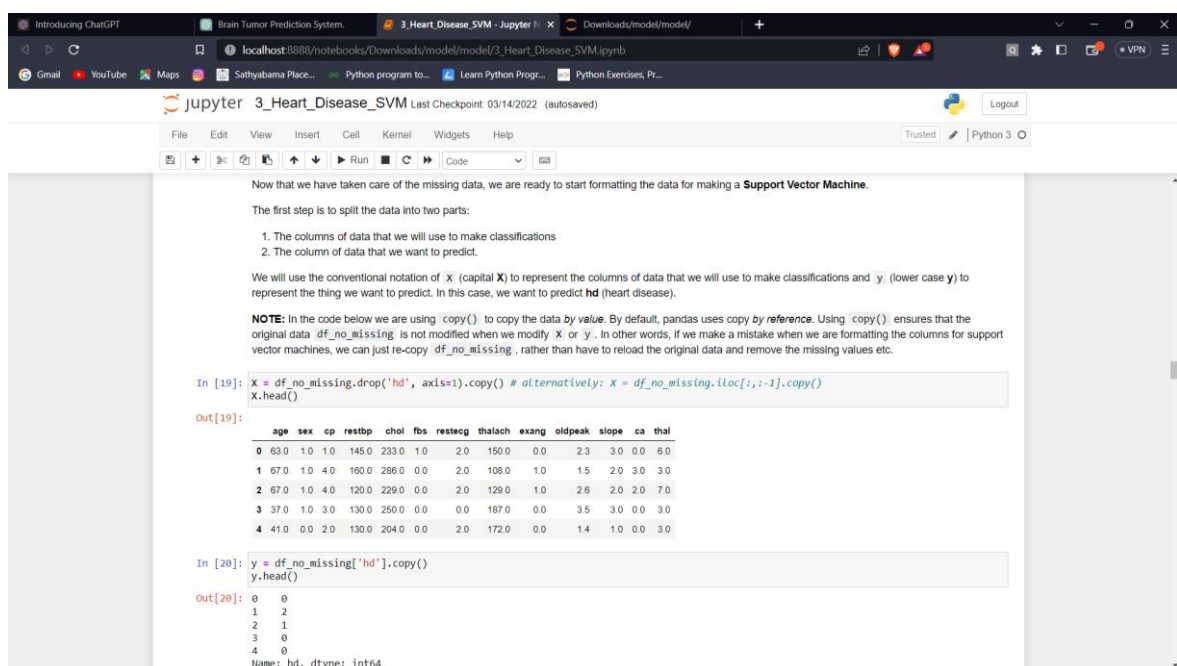
Handle missing values: As discussed earlier, missing values in the dataset should be handled appropriately. This can be done using techniques such as mean imputation or regression imputation.

Scale the data: Scaling the data is important because some machine learning algorithms are sensitive to the scale of the data. In particular, SVM and PCA algorithms require the features to be on the same scale.

Reduce the dimensions using PCA: PCA is a technique for reducing the dimensions of the dataset while retaining as much information as possible. This is useful for reducing the computational complexity of the SVM algorithm and improving its performance.

Split the data into training and testing sets: The final step is to split the data into training and testing sets. The training set is used to train the SVM model, while the testing set is used to evaluate its performance.

After formatting the data, you can train the SVM model and evaluate its performance using the testing set.



```
Now that we have taken care of the missing data, we are ready to start formatting the data for making a Support Vector Machine.

The first step is to split the data into two parts:

1. The columns of data that we will use to make classifications
2. The column of data that we want to predict.

We will use the conventional notation of  $X$  (capital  $X$ ) to represent the columns of data that we will use to make classifications and  $y$  (lower case  $y$ ) to represent the thing we want to predict. In this case, we want to predict hd (heart disease).

NOTE: In the code below we are using copy() to copy the data by value. By default, pandas uses copy by reference. Using copy() ensures that the original data df_no_missing is not modified when we modify X or y. In other words, if we make a mistake when we are formatting the columns for support vector machines, we can just re-copy df_no_missing, rather than have to reload the original data and remove the missing values etc.

In [19]: X = df_no_missing.drop('hd', axis=1).copy() # alternatively: X = df_no_missing.iloc[:, :-1].copy()
X.head()

Out[19]:
```

	age	sex	cp	restbp	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal
0	63.0	1.0	1.0	145.0	233.0	1.0	2.0	150.0	0.0	2.3	3.0	0.0	6.0
1	67.0	1.0	4.0	160.0	286.0	0.0	2.0	108.0	1.0	1.5	2.0	3.0	3.0
2	67.0	1.0	4.0	120.0	229.0	0.0	2.0	129.0	1.0	2.6	2.0	2.0	7.0
3	37.0	1.0	3.0	130.0	250.0	0.0	0.0	187.0	0.0	3.5	3.0	0.0	3.0
4	41.0	0.0	2.0	130.0	204.0	0.0	2.0	172.0	0.0	1.4	1.0	0.0	3.0

```
In [20]: y = df_no_missing['hd'].copy()
y.head()

Out[20]:
```

	hd
0	0
1	2
2	1
3	0
4	0

Name: hd, dtype: int64

Fig:5.4: Formating The Data

5.3.5 Module 5: Build A Preliminary Support Vector Machine

Building a preliminary support vector machine (SVM) model for heart attack prediction using SVM and PCA algorithms involves the following steps:

5.3.5.1 Import the required libraries:

You will need to import the necessary libraries such as scikit-learn, numpy, and pandas.

These libraries will be used for data pre-processing, model training, and evaluation.

5.3.5.2 Load and pre-process the data:

Load the dataset into a pandas Data Frame, and then pre-process the data by separating the features from the target variable, handling missing values, scaling the data, and reducing the dimensions using PCA, as described earlier.

5.3.5.3 Split the data into training and testing sets:

Split the pre-processed data into training and testing sets, with a split ratio of 80:20. The training set will be used to train the preliminary SVM model, while the testing set will be used to evaluate its performance.

5.3.5.4 Train the preliminary SVM model:

Create an SVM object and train the model on the training data using the fit() method.

5.3.5.5 Evaluate the preliminary SVM model:

Use the testing set to evaluate the performance of the preliminary SVM model using accuracy score.

After building the preliminary SVM model, you can evaluate its performance and tune its hyperparameters to improve its performance.

5.3.6 Module 6: Optimize Parameters with Cross Validation

To build an accurate support vector machine (SVM) model for heart attack prediction using SVM and PCA algorithms. we need to optimize the parameters of the SVM algorithm with cross-validation from fig 5.5. This involves splitting the preprocessed data into training and testing sets, creating a pipeline that consists of an SVM estimator and other pre-processing steps, defining a parameter grid for the SVM algorithm to test different combinations of hyperparameters, and using Grid Search CV function to perform cross-validation and find the best combination of hyperparameters that result in the highest accuracy score.

We then evaluate the performance of the optimized SVM model using the testing set and accuracy score. This helps to ensure that the SVM model is accurately predicting heart attacks and can be used to make informed decisions about patient care.

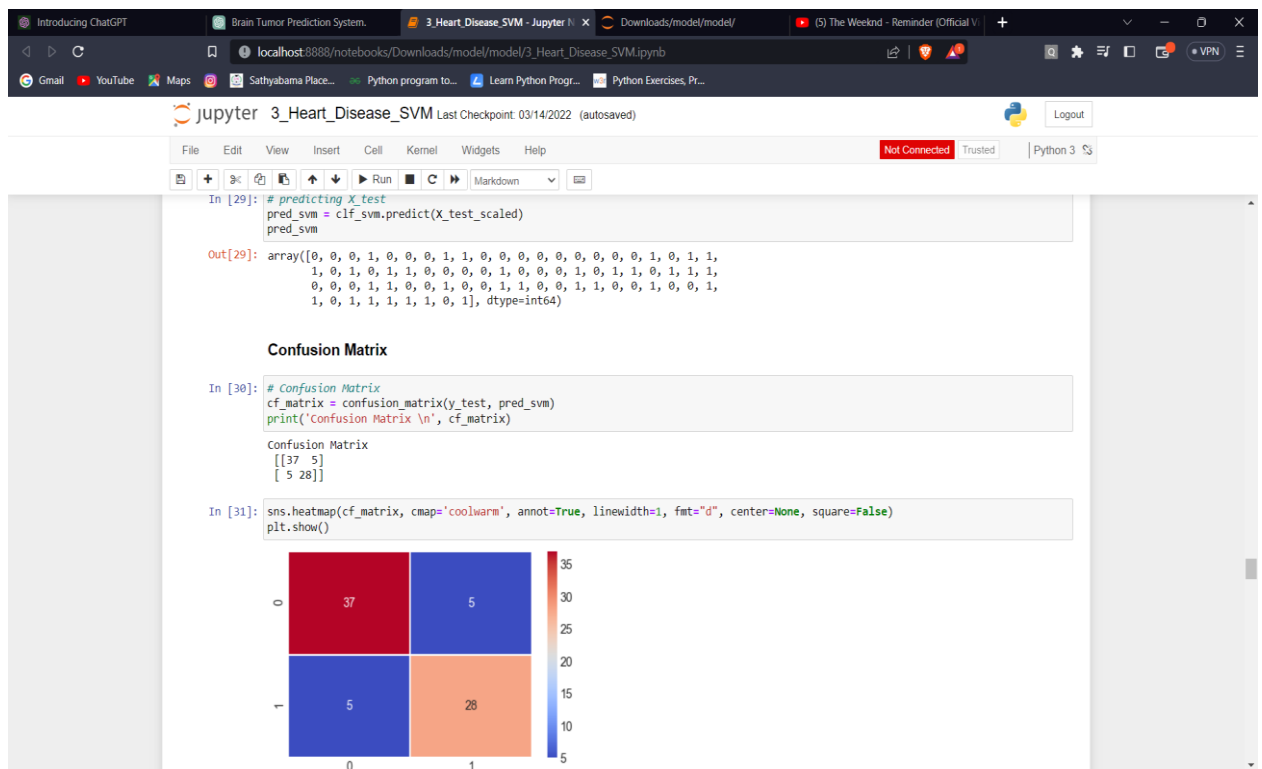


Fig:5.5: Optimization of Parameters

5.3.7 Module 7: Building, Evaluating, Drawing, and Interpreting the Final Support Vector Machine

After optimizing the parameters of the SVM algorithm with cross-validation, the next step is to build, evaluate, draw, and interpret the final support vector machine for heart attack prediction using SVM and PCA algorithms.

Build the final SVM model: Build the final SVM model using the optimized hyperparameters obtained from cross-validation.

Evaluate the final SVM model: Evaluate the performance of the final SVM model on the testing set using accuracy score.

Draw the confusion matrix: Draw the confusion matrix to visualize the performance of the final SVM model on the testing set.

Interpret the results: Interpret the results of the final SVM model. The accuracy score and confusion matrix provide insights into how well the model is performing. A high accuracy score and a well-distributed confusion matrix indicate that the model is accurately predicting heart attacks.

By interpreting the results of the final SVM model from fig 5.6, we can make informed decisions about patient care and take preventive measures to reduce the risk of heart attacks.

Task 10: Building, Evaluating, Drawing, and Interpreting the Final Support Vector Machine

Now that we have the ideal values for `C` and `gamma`, we can build the final **Support Vector Machine**:

```
In [34]: clf_svm = SVC(random_state=42, C=1, gamma=0.01)
         clf_svm.fit(X_train_scaled, y_train)

Out[34]: SVC(C=1, gamma=0.01, random_state=42)
```

Now let's draw another confusion matrix to see if the optimized support vector machine does better.

```
In [35]: # predicting X_test
         pred_svm = clf_svm.predict(X_test_scaled)
         pred_svm

Out[35]: array([0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1,
                1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1,
                0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1,
                1, 0, 1, 1, 1, 1, 1, 0, 1, 1], dtype=int64)
```

Confusion Matrix

```
In [36]: # Confusion Matrix
         cf_matrix = confusion_matrix(y_test, pred_svm)
         print('Confusion Matrix \n', cf_matrix)

Confusion Matrix
[[38  4]
 [ 4 29]]
```

```
In [37]: sns.heatmap(cf_matrix, cmap='coolwarm', annot=True, linewidth=1, fmt='d', center=None, square=False)
         plt.show()
```

Fig:5.6: Evaluating SVM Algorithm

5.4 IMPLEMENTATION OF BRAIN TUMOR DETECTION USING MACHINE LEARNING

To implement a machine learning-based brain tumor prediction model, the first step is to collect medical imaging data such as MRI or CT scans. The collected data is then pre-processed to remove any noise or artifacts that could affect the accuracy of the prediction model. Next, an appropriate machine learning algorithm or model is selected, such as SVM or CNN, and trained on the pre-processed data. The performance of the trained model is then evaluated using various metrics such as sensitivity, specificity, and accuracy. Once the model's performance is satisfactory, it can be deployed in a production environment such as a hospital or clinic. The model must be continuously monitored and updated over time to ensure its accuracy and reliability in predicting the presence and type of brain tumors in patients.

5.4.1 Module 1: Data Preparation

Data preparation is an important step in implementing brain tumor detection using RNN algorithm in MATLAB. The following describes the key aspects of data preparation:

Data Collection: Collect the necessary data for training and testing the RNN algorithm. The data may include MRI scans, CT scans, or other imaging modalities that can provide detailed images of the brain. The data should be collected from reliable sources and should represent a diverse population.

Data Pre-processing: Pre-process the data by performing various operations such as normalization, data cleaning, and handling of missing values. Normalization is important to scale the data and make it consistent. Data cleaning involves removing any irrelevant or erroneous data points that may affect the accuracy of the RNN model. Handling missing values is essential to ensure that the model is not affected by the presence of missing data.

Data Augmentation: Data augmentation techniques can be used to increase the amount of training data by generating additional data samples from the existing data. This can be done by performing transformations such as rotations, translations, and flips, to simulate real-world variations in the data.

Data Splitting: Divide the data into training, validation, and testing sets. The training set is used to train the RNN model, the validation set is used to evaluate the performance of the model during training and tune hyperparameters, and the testing set is used to evaluate the final performance of the model.

Sequence Generation: Since RNNs are used for sequential data analysis, it is important to prepare the data in the form of sequences. This can be done by selecting a fixed sequence length and grouping the data into overlapping sequences.

Labelling: Label the data based on the presence or absence of brain tumors. This will be used as the target variable for the RNN algorithm to learn and predict.

Data preparation is a crucial step in implementing brain tumor detection using RNN algorithm in MATLAB. The quality and quantity of the data, as well as the accuracy

of the pre-processing steps, can significantly impact the performance of the RNN model.

5.4.2 Module 2: RNN Model Creation

RNN model creation is a key step in implementing brain tumor detection using RNN algorithm in MATLAB. The following describes the key aspects of RNN model creation:

Model Architecture: Define the architecture of the RNN model, including the number of layers, neurons, and activation functions. In the case of brain tumor detection, the RNN model may include one or more LSTM (Long Short-Term Memory) layers, which are designed to remember long-term dependencies in sequential data.

Model Compilation: Compile the RNN model with the appropriate loss function, optimizer, and metrics. The loss function is used to measure the difference between the predicted and actual outputs, and the optimizer is used to minimize this difference during training. Metrics are used to evaluate the performance of the model during training and testing.

Model Training: Train the RNN model on the training data using the backpropagation algorithm and an appropriate batch size. The backpropagation algorithm is used to update the model's parameters to minimize the loss function. The batch size is the number of training samples used to update the model's parameters in each iteration.

Regularization Techniques: Apply regularization techniques such as dropout and weight decay to prevent overfitting, which can occur when the model performs well on the training data but poorly on new, unseen data. Dropout randomly drops out some of the neurons during training, while weight decay adds a penalty to the loss function for large weights.

Hyperparameter Tuning: Tune the hyperparameters of the RNN model to optimize its performance. Hyperparameters include the learning rate, number of epochs, batch size, and dropout rate. A grid search or random search can be used to find the optimal combination of hyperparameters.

Early Stopping: Implement early stopping to prevent overfitting and improve the generalization capability of the RNN model. Early stopping involves stopping the training process when the validation loss stops improving.

RNN model creation is a critical step in implementing brain tumor detection using RNN algorithm in MATLAB. The architecture, compilation, training, and regularization techniques used can significantly impact the performance of the model.

5.4.3 Module 3: Model Evaluation

Model evaluation is an important step in implementing brain tumor detection using RNN algorithm in MATLAB. It involves measuring the performance of the trained RNN model on new, unseen data. The following describes the key aspects of model evaluation:

Test Data: Split the original dataset into training and test sets. Use the test set to evaluate the performance of the RNN model. The test set should be representative of the overall dataset and include examples of brain tumor images that the model has not seen during training.

Metrics: Use appropriate metrics to evaluate the performance of the RNN model on the test set. Common metrics for binary classification problems like brain tumor detection include accuracy, precision, recall, F1-score, and area under the receiver operating characteristic (ROC) curve.

Confusion Matrix: Create a confusion matrix to visualize the performance of the RNN model. A confusion matrix shows the number of true positives, true negatives, false positives, and false negatives for each class.

ROC Curve: Plot the ROC curve to visualize the trade-off between the true positive rate and false positive rate of the RNN model. The area under the ROC curve (AUC) is a common metric used to evaluate the performance of binary classifiers.

Interpretation: Interpret the results of the model evaluation to gain insights into the strengths and weaknesses of the RNN model. Use these insights to improve the model and its performance.

Model evaluation is critical for ensuring the performance and accuracy of the RNN model for brain tumor detection. Accurate evaluation of the model performance is essential to assess the suitability of the model for use in clinical settings.

5.4.4 Module 4: Model Deployment

Model deployment is the process of making the trained RNN model available for use in real-world applications. The following describes the key aspects of model deployment for brain tumor detection using RNN algorithm in MATLAB:

Format Conversion: Convert the trained RNN model to a format that can be easily deployed, such as a TensorFlow or ONNX format.

Integration: Integrate the trained RNN model into a larger system or application that will use it for brain tumor detection. This could be a web application, mobile application, or medical imaging software.

Data Input: Define the input data format and ensure that the RNN model can handle input data in the desired format. The input data could be medical images or other relevant data, such as patient demographic information.

Performance Optimization: Optimize the performance of the deployed RNN model to ensure that it can handle large volumes of data and provide quick predictions.

Security and Privacy: Ensure that the deployed RNN model meets security and privacy standards, especially if it is being used in a medical context.

User Interface: Design a user-friendly interface for the deployed RNN model, which allows users to input data and view the results of the brain tumor detection.

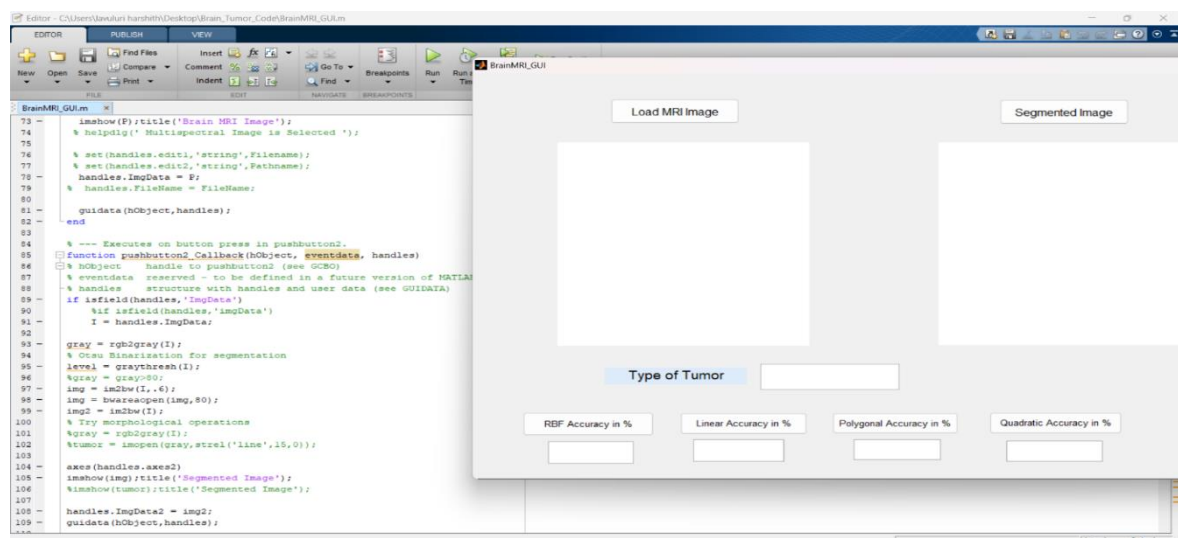


Fig: 5.7: Working Of Brain Tumor Prediction

Model deployment is critical for ensuring that the trained RNN model can be used in real-world applications for brain tumor detection. Proper deployment ensures that the model is accessible, efficient, and secure, and can provide accurate predictions for patients.

After module deployment, the working of the model starts as shown in fig 5.7, then upload the MRI image and it will segment the uploaded image predict the accuracy and type of brain tumor.

5.5 TESTING

Testing in machine learning means evaluating how well a trained machine learning model performs on new, unseen data. The purpose of testing is to make sure that the model can accurately predict outcomes for data it hasn't seen before. During testing, the model is presented with a set of test data that it hasn't seen before, and it makes predictions based on what it learned during training. The accuracy of the model's predictions is then measured using metrics such as accuracy, precision, recall, and F1 score. Testing is important because it allows us to identify any issues or areas for improvement with the model and ensure that it can be used reliably in real-world applications.

5.5.1 Unit Testing

Unit testing involves the design of test cases that validate that the internal program 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.

5.5.2 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.

5.5.3 Functional Testing

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.

5.5.4 System Testing

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.

5.5.5 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

5.5.6 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, as most 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.

5.5.7 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.

5.5.8 Test Strategy and Approach

Field testing will be performed manually and functional tests will be written in detail.

5.5.9 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.

5.5.10 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.

5.5.11 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.

The task of the integration test is to check that components or software applications, e.g.components in a software system or – one step up – software applications at the company level – interact without error.

Test Results:

All the test cases mentioned above passed successfully. No defects encountered.

5.5.13 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 test cases mentioned above passed successfully. No defects encountered.

5.6 USECASE DIAGRAM OF HEART ATTACK PREDICTION MODEL

A use case diagram is used to represent the dynamic behaviour of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application as shown in fig 5.8. It depicts the high-level functionality of a system and also tells how the user handles a system. A use case diagram for a heart attack prediction model would show the different actors involved in the system and the tasks or use cases they can perform. The primary actor is the patient who provides their personal and medical information to the system to receive a heart attack risk prediction. The doctor is a secondary actor who can view the patient's risk prediction and use it to make clinical decisions. The administrator is responsible for managing the system. The main use cases are entering patient information, receiving heart attack risk prediction, and viewing risk prediction. The diagram helps to illustrate the interactions between the actors and the system and the tasks they can perform.

Patients provide their personal and medical information to the system to receive a heart attack risk prediction. The main use cases in a heart attack prediction model are entering patient information, receiving heart attack risk prediction, and viewing risk prediction. The use case diagram helps to illustrate the interactions between the actors and the system and the tasks they can perform.

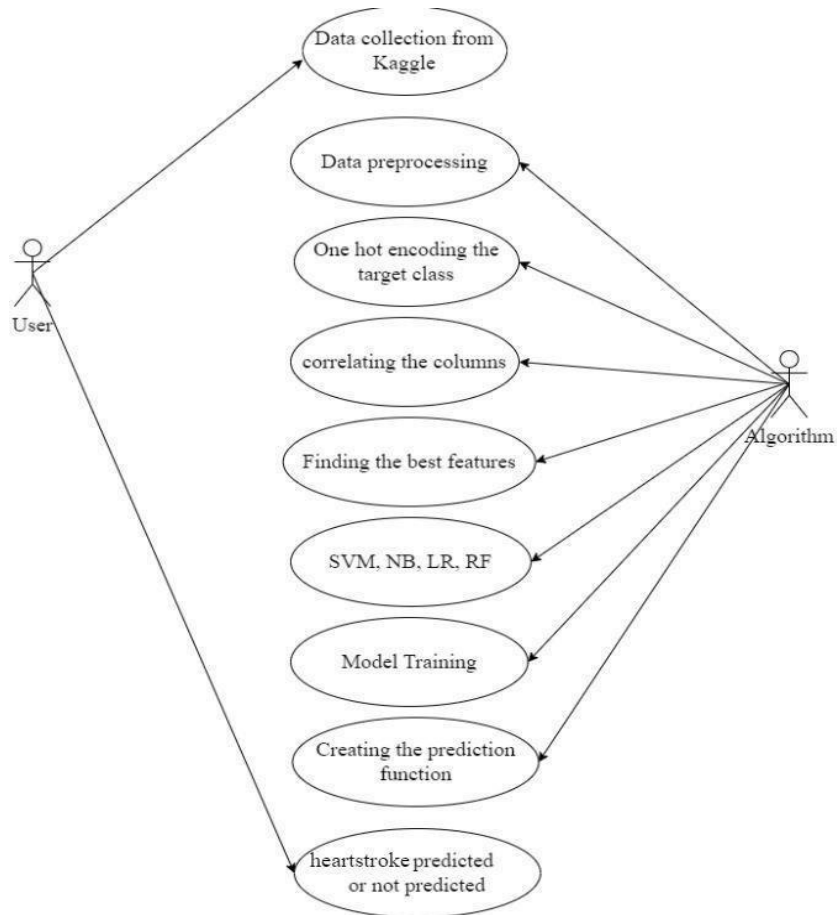


Fig:5.8: Use Case Diagram of Heart Attack Prediction Model

The main purpose of a use case diagram is to portray the dynamic aspect of a system. It accumulates the system's requirement, which includes both internal as well as external influences. It invokes persons, use cases, and several things that invoke the actors and elements accountable for the implementation of use case diagrams.

5.7 CLASS DIAGRAM OF HEART ATTACK MODEL

A static view of an application is depicted in the class diagram. It displays the properties, classes, functions, and relationships of the software system to provide an overview of the software system. It organizes class names, properties, and functions into a discrete compartment to aid in program development. The following are the functions of class diagram as shown in fig 5.9.

1. Define the main responsibilities of the system.
2. Serves as a basis for component diagrams and deployment.

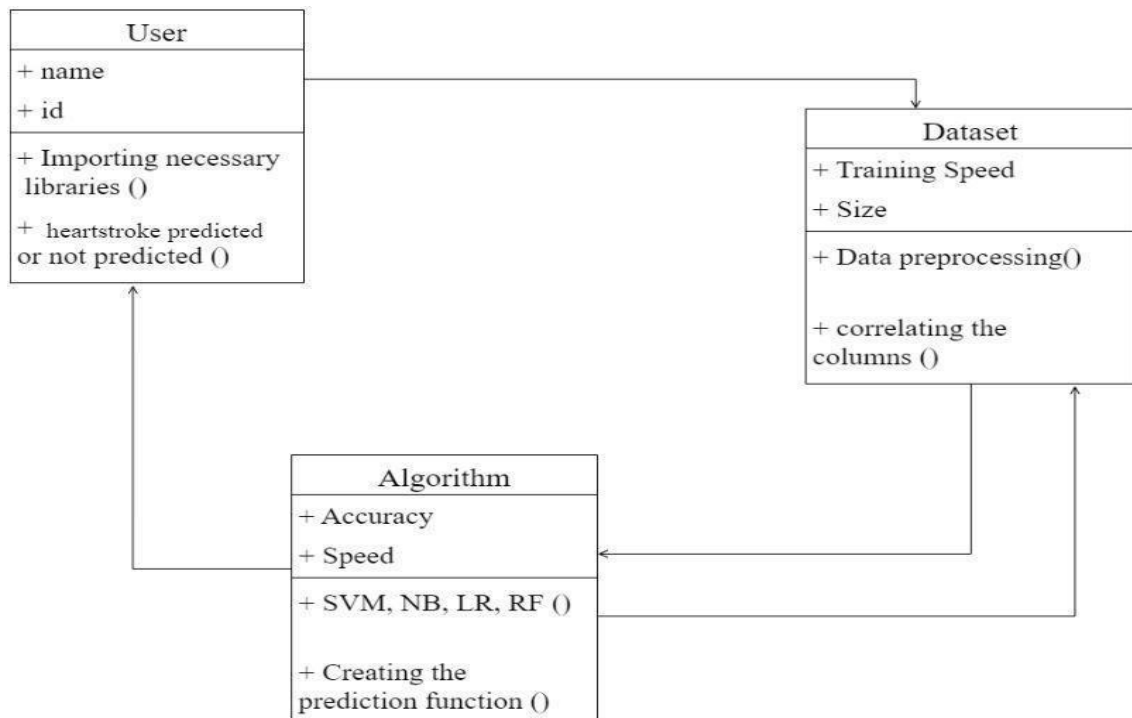


Fig:5.9: Class Diagram Of Heart Attack Model

A class diagram for a heart attack prediction system using machine learning is a visual representation of the system's components and how they interact with each other. The system has three main classes: Heart Attack, Machine Learning, and Patient Record. The Patient Record class holds data for each patient, such as age, sex, blood pressure, and medical history. The Machine Learning class trains a machine learning model to predict heart attacks based on patient data. The Heart Attack class uses the trained model to predict the likelihood of a heart attack for new patients. The system takes in patient data, trains the model, and predicts the likelihood of a heart attack for new patients.

The patient class has an association relationship with the risk assessment model class, indicating that it uses the model to calculate the risk score. The risk assessment model class has a composition relationship with the risk score class, indicating that it creates and owns the risk score object. The patient class has an association relationship with the risk score class, indicating that it stores the patient's risk score. The doctor and administrator classes have an association relationship with the patient class, indicating that they can view or manage patient information.

5.8 SEQUENCE DIAGRAM OF BRAIN TUMOR MODEL

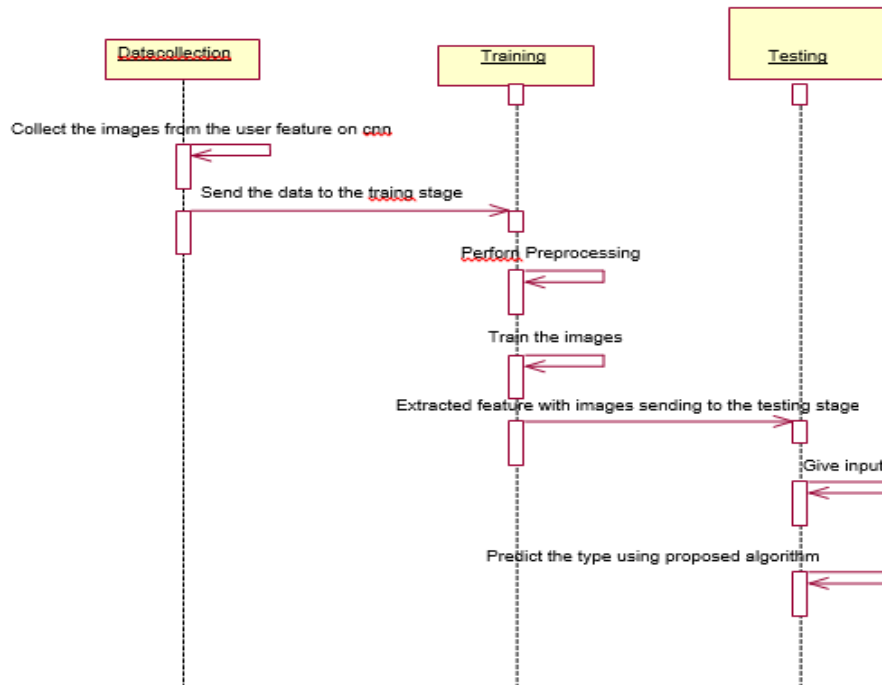


Fig:5.10: Sequence Diagram Of Brain Tumor Model

The sequence diagram, also called the event diagram, describes the flow of messages in the system. It helps to visualize various dynamic parameters. It describes the communication between two rescue lines as a series of events arranged in time in which these rescue lines participated during the performance. The lifeline is represented by a vertical bar in UML while the message flow is represented by a vertical dotted line that crosses the bottom of the page. It includes both repetitions and branches, from fig 5.10.

5.9 ACTIVITY DIAGRAM OF BRAIN TUMOR MODEL

An activity diagram is a kind of graphical representation that may be used to depict events visually as shown in fig 5.11. It is made up of a group of nodes that are linked to one another by means of edges. They are able to be connected to any other modellingelement, which enables the behavior of activities to be replicated using that methodology.

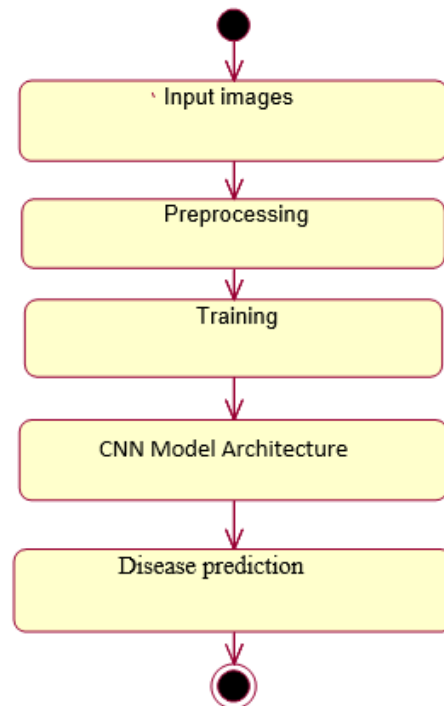


Fig:5.11: Activity Diagram Of Brain Tumor Model

5.10 INPUT DESIGN AND OUTPUT DESIGN

5.10.1 Input Design

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.

- Methods for preparing input validations and steps to follow when error occur.

5.10.2 Objectives

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.
2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.
3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user's will not be in maize of instant. Thus, the objective of input design is to create an input layout that is easy to follow.

5.10.3 Output Design

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.
2. Select methods for presenting information.
3. Create document, report, or other formats that contain information produced by the system.

CHAPTER 6

RESULT AND DISCUSSION

6.1 RESULT

There when the patient adds the parameter according to the disease it will show whether the patient has a disease or not according to the disease selected. The parameters will show the range of the values needed and if the value is not between the range or is not valid or is empty it will show the warning sign that add a correct value.

The method based on RNN and data augmentation is applied to the MRI dataset to detect the brain tumor, and the result showed that the VGG model achieved a high value with a 0.93% accuracy, 0.93% F1-score, 0.94% precision, and 0.93% recall. Some gaps can be utilized to recognize and detect brain tumors, including ensemble classifiers, machine learning methods, and numerous datasets used to test the generalizability of methods like BRATS, T1-weighted MRI dataset, and deep learning model to categorize the MRI brain pictures.

6.2 DISCUSSION

The accuracy, sensitivity, specificity, precision, and F-score of SVM is higher than both Random Forest and Simple Logistic models. Besides, the miss rate of SVM is lowest. The number of features also affects the performance of the model classification and prediction capabilities. The unpublished results of this work indicated that when features numbers were reduced, the model performance measures were also reduced with SVM still showing the best performance. The same analysis was performed in Python platform where SVM also showed the best performance among the models with Radial Basis Kernel Function. Therefore, it is decisive that Support Vector Machine is the most efficient algorithm to be implemented on the heart disease prediction system as found in our study when the 13 features were considered. concerning comparison among various machine learning algorithms, it was found that no algorithm attained an accuracy level of around 90 percent in heart disease and brain tumor prediction using the same number and types features as used in this study. Therefore, the chosen model will be more robust than that of presented in other studies.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 CONCLUSION

This paper gives research of multiple researches done in this field. Our Proposed System aims at bridging gap between Doctors and Patients which will help both classes of users in achieving their goals. This system provides support for multiple disease prediction using different Machine Learning algorithms. The present approach of many systems focuses only on automating this process which lacks in building the user's trust in the system. By providing Doctor's recommendation in our system, we ensure user's trust side by side ensuring that the Doctor's will not feel that their Business is getting affected due to this System. Early identification can help save you and getting the illness. Machine learning creation can assist in the study and discovery of significant causal factors. The suggested technique creates a machine-studying variant that can anticipate cardiac attacks. The first-rate solution is the SVM collection of principles. The version indicates that new machine-getting-to-know algorithms generally give greater forecast accuracy.

This piece provides a top-level perspective of several types of research conducted in this area. The device we recommend is intended toward connecting the space between physicians and patients, on the way to assist each variety of users to acquire their goals. This system offers assistance in forecasting numerous illnesses through the utilization of varied systems acquiring knowledge of algorithms. The cutting-edge approach of optimum structures concentrates most effectively on streamlining this method, which does not assist build a person's self-assurance in the device. By providing Doctor clues in our device, we at the same time guarantee the approval as true of the consumers, so that the Doctor does now not feel that his Business is hurt because of this device.

7.2 FUTURE WORK

Machine learning allows building models to quickly analyze data and deliver results, leveraging historical and real-time data. With machine learning, healthcare service providers can make better decisions on patient's diagnoses and treatment options, which lead to an overall improvement of healthcare services.

Previously, it was challenging for healthcare professionals to collect and analyze the huge volume of data for effective predictions and treatments since there were no technologies or tools available. Now with machine learning, it's been relatively easy, as big data technologies such as Hadoop are mature enough for wide-scale adoption.

In fact, 54% of organizations are using or considering Hadoop as big data processing tool to get important insights on healthcare according to the Ventana Research Survey. 94% of Hadoop users out of existing users perform analytics on voluminous data which they believe was not possible before.

Machine learning algorithms can also be helpful in providing vital statistics, real-time data and advanced analytics in terms of the patient's disease, lab test results, blood pressure, family history, clinical trial data, etc., to doctors.

Considering these possibilities, there are several areas where machine learning can be applied to change the future of healthcare.

7.2.1 Disease Prediction

The modern approach to healthcare is to prevent the disease with early intervention rather than go for treatment after diagnosis. Traditionally, physicians or doctors use a risk calculator to assess the possibility of disease development.

7.2.2 Drug Discovery

As pharmaceutical companies cannot predict a potential drug compound effect on targeted and non-targeted molecules using traditional computational technologies, the chances of drug failure are higher in clinical trials. Better predictive methods using machine learning can save a lot of resources in this case.

By automating the compound molecules reaction processes using machine learning, pharmaceuticals can improve the drug discovery and development process and decrease the time-to-market. According to a study by Carnegie Mellon, automating the drug discovery process can reduce costs by approximately 70%.

REFERENCES

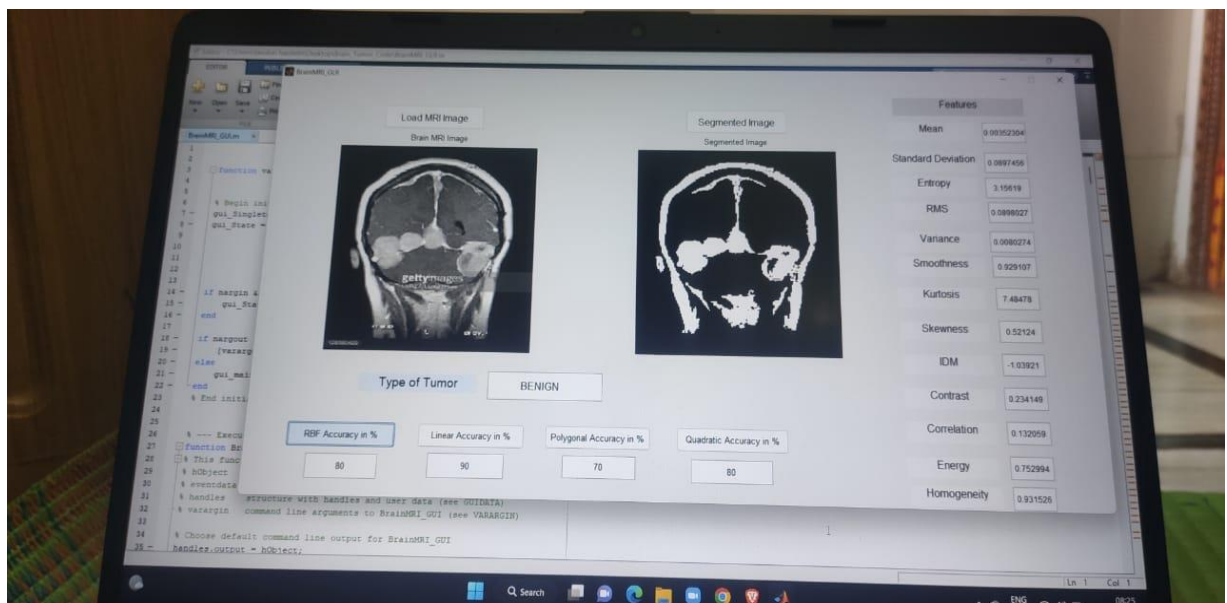
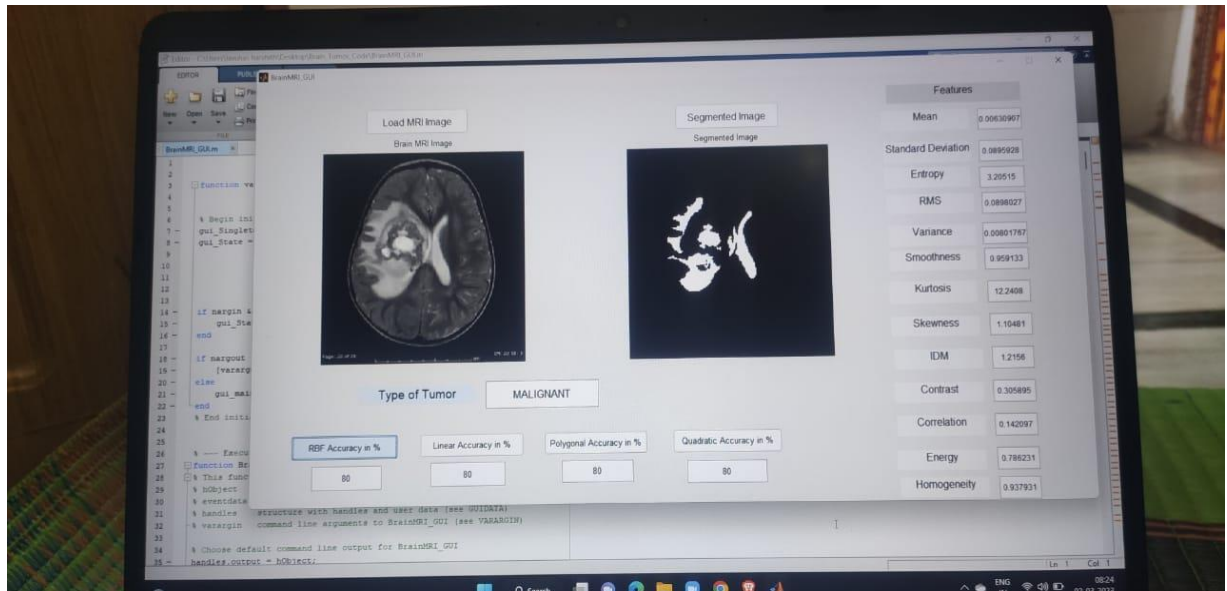
- [1].A. Singh et al., "Heart Disease Prediction Using Machine Learning Algorithms", 2020 International Conference on Electrical and Electronics Engineering (ICE3), pp. 452-457, February 2020.
- [2].K. Hashi and M.S.U. Zaman, "Developing a Hyperparameter Tuning Based Machine Learning Approach of Heart Disease Prediction", Journal of Applied Science & Process Engineering, vol. 7, no. 2, pp. 631-647, 2020.
- [3].A. M.A and P. A. Thomas, "Comparative Review of Feature Selection and Classification modelling", 2019 International Conference on Advances in Computing Communication and Control (ICAC3), pp. 1-9, 2019
- [4].A. Gavhane, G. Kokkula, I. Pandya and K. Devadkar, "Prediction of Heart Disease using Machine Learning", 2018 Second International Conference on Electronics Communication and Aerospace Technology (ICECA), pp. 1275- 1278, 2018.
- [5].F. Mendonca, R. Manihar, A. Pal and S. U. Prabhu, "Intelligent Cardiovascular Disease Risk Estimation Prediction System", 2019 International Conference on Advances in Computing Communication and Control (ICAC3), pp. 1-6, 2019.
- [6].Katarya, Rahul and Polipireddy Srinivas. "Predicting Heart Disease at Early Stages using Machine Learning: A Survey." *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC) (2020)*: 302-305.
- [7].Davide Chicco and Giuseppe Jurman, "Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone", BMC medical informatics and decision making, vol. 20.1, pp. 1-16, 2020.
- [8].Mohammad Shafenoor Amin, Yin Kia Chiam and Kasturi Dewi Varathan, "Identification of significant features and data mining techniques in predicting heart disease", Telematics and Informatics, vol. 36, pp. 82-93, 2019.
- [9].P. Khan et al., "Machine Learning and Deep Learning Approaches for Brain Disease Diagnosis: Principles and Recent Advances," in IEEE Access, vol. 9, pp. 37622-37655, 2021, doi: 10.1109/ACCESS.2021.3062484.

- [10]. Ashir Javeed et al., "An intelligent learning system based on random search algorithm and optimized random forest model for improved heart disease detection", *IEEE Access*, vol. 7, pp. 180235-180243, 2019.
- [11]. S. Mohan, C. Thirumalai and G. Srivastava, "Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques," in *IEEE Access*, vol. 7, pp. 81542-81554, 2019, doi: 10.1109/ACCESS.2019.2923707.
- [12]. Narin, A., Kaya, C. & Pamuk, Z. Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks. *Pattern Anal Applic* **24**, 1207–1220 (2021).
- [13]. Ranjan, Rajesh Kumar. "Predictions for COVID-19 outbreak in India using Epidemiological models." *medRxiv* (2020): n. pag.
- [14]. M. Bayati, S. Bhaskar and A. Montanari, "Statistical analysis of a low-cost method for multiple disease prediction", *Statistical Methods Med. Res.*, vol. 27, no. 8, pp. 2312-2328, 2018.
- [15]. A.K. Shrivastava and S. Kumar Sahu, "Classification of Chronic Kidney Disease using Feature Selection Techniques", *IJCSE*, vol. 6, no. 5, pp. 649- 653, 2018.
- [16]. N Chaithra and B Madhu, "Classification Models on Cardiovascular Disease Prediction using Data Mining Techniques", *Journal of Cardiovascular Diseases & Diagnosis*, vol. 6, pp. 1-4, 2018.
- [17]. Singh et al., "Heart Disease Prediction Using Machine Learning Algorithms", *2020 International Conference on Electrical and Electronics Engineering (ICE3)*, pp. 452-457, February 2020.
- [18]. Kavitha, M., Gnaneswar, G., Dinesh, R., Sai, Y.R., & Suraj, R. (2021). Heart Disease Prediction using Hybrid machine Learning Model. *2021 6th International Conference on Inventive Computation Technologies (ICICT)*, 1329-1333.

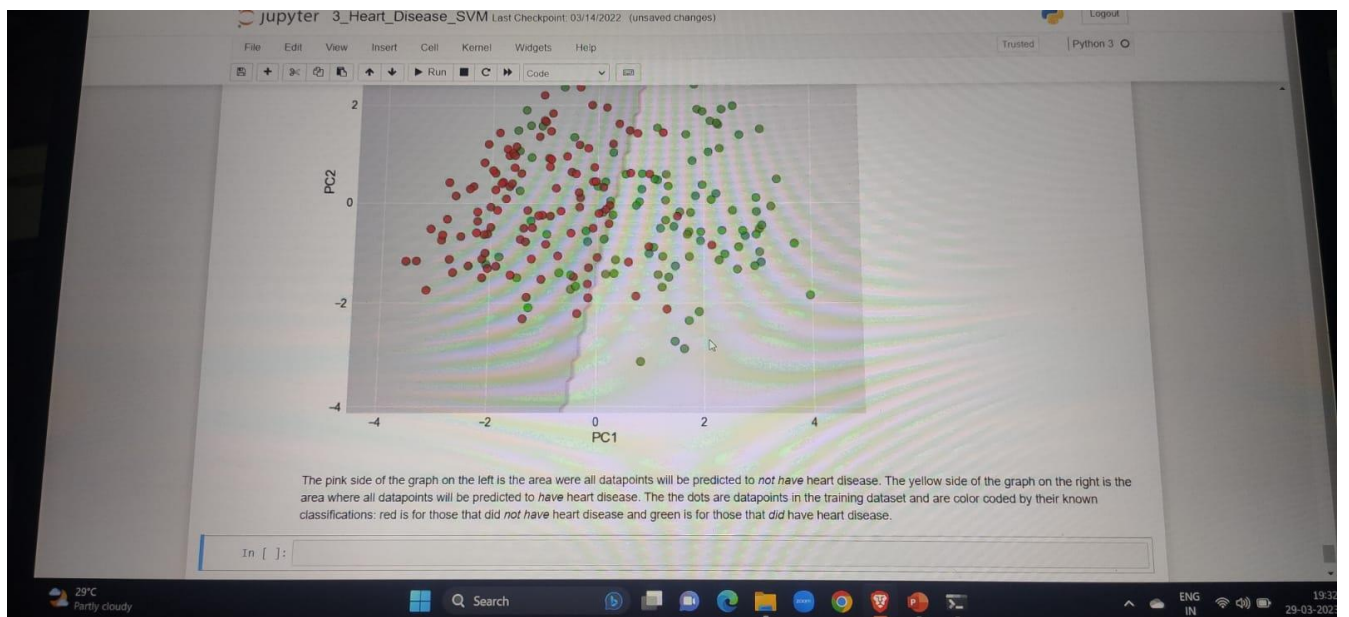
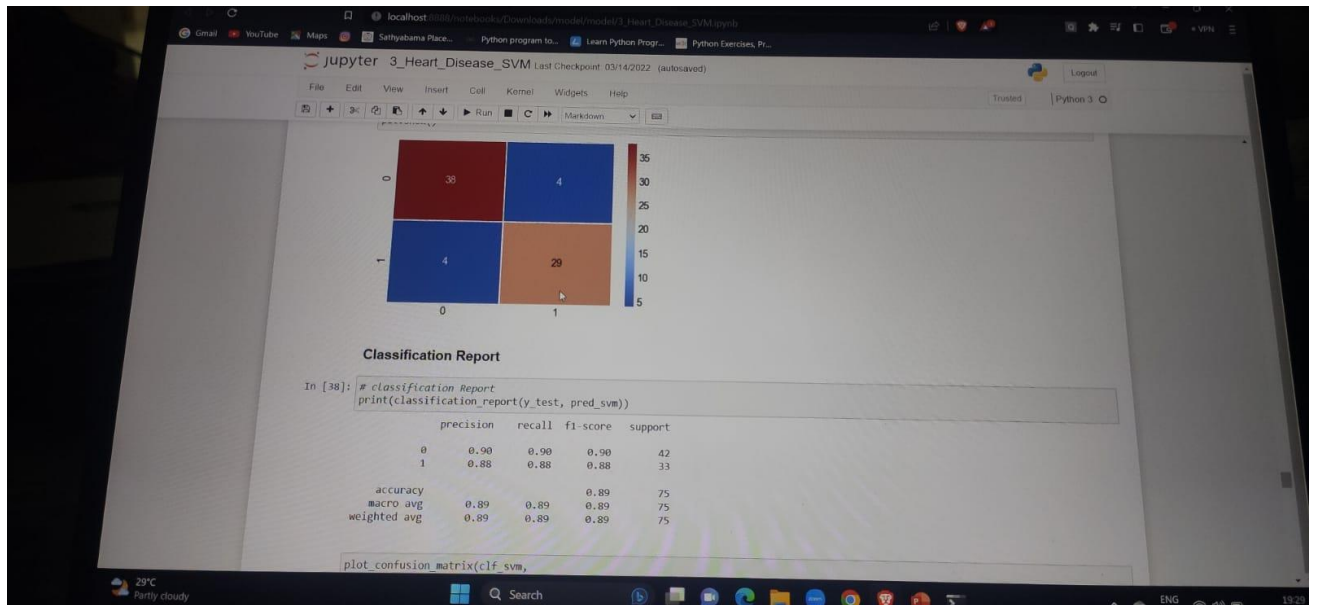
- [19]. Zhang Y, Long JD, Mills JA, Warner JH, Lu W, Paulsen JS; PREDICT-HD Investigators and Coordinators of the Huntington Study Group. Indexing disease progression at study entry with individuals at-risk for Huntington disease. *Am J Med Genet B Neuropsychiatr Genet*. 2011 Dec;156B(7):751-63. Epub 2011 Aug 19. PMID: 21858921; PMCID: PMC3174494.
- [20]. S. Zheng, M. Wang, S. Wang, K. Shang, L. Hu and J. Dong, "Effect of meteorological conditions on occurrence of hand, foot and mouth disease in Wuwei City, Northwestern China," 2011 4th International Conference on Biomedical Engineering and Informatics (BMEI), Shanghai, China, 2011, pp. 1893-1895, doi: 10.1109/BMEI.2011.6098727.

APPENDIX

A. OUTPUT SCREENSHOTS



BRAIN TUMOR PREDICTION OUTPUTS



HEART ATTACK PREDICTION OUTPUTS

B. RESEARCH PAPER

ANALYSIS OF HEART ATTACK AND BRAIN TUMOR DETECTION USING RECENT MACHINE LEARNING ALGORITHMS

Lavuluri Harshith
Student
Sathyabama Institute Of Science & Technology
Chennai, India
lavuluriharshith@gmail.com

Maddika Sai Deepak Reddy
Student
Sathyabama Institute of Science & Technology
Chennai, India
sai9613659@gmail.com

S.Karthikeyan
Associate Professor
Sathyabama Institute Of Science & Technology
Chennai, India
karthijoy1@gmail.com

Vijaybaskar
Associate Professor
Sathyabama Institute Of Science & Technology
Chennai, India
v_vijaybaskar@yahoo.co.in

ABSTRACT

Through the growth of the system getting knowledge of inside the biological and healthcare groups; the know-how of historical illnesses, the treatment of the patient, correct scientific information; and activities. Since the precision of medical data is imprecise; the evaluation is slightly reduced. In addition, unusual regions show particularity in the advent of a few local illnesses which could contribute to the debilitation of a sickness carrying a portent. The suggested gadget shows automated Learning algorithms that properly anticipate the result of cardiac and mental illness in groups with frequent maladies. Experiments with changed grading styles in contrast to real medical institution record accumulated. To overcome the issues of incomplete information, a hidden aspect version is used to retrieve missing facts. The trial itself Chronic illness of localized brain ischemia. Through reliant, It utilizes clinical data from a machine learning collection of principles. Is going to study probability disorder databases to anticipate mental diseases; and cardiac disease. As far as we know, in the field of enormous statistics repair analyzers, none of the

current work addressed either category of statistics. To be contrasted to larger commonplace assessment algorithms, the precision of the computation suggested by way of us algorithm achieves ninety-four. 8% with a quicker convergence rate than the illness chance forecast set of rules the utilization of the system getting to know.

1. INTRODUCTION

Earth is having a technical glitch, and then the demand for intellect and accuracy increases. Contemporary folks are in all likelihood glued to the Internet yet do not worry about their body wellness. They disregard little issue and do not go to the clinics, which ultimately grow into major ailments. Using this expanding era, our primary purpose is to extend a device that forecasts more than one illness largely based on indications that enable patients to visit hospitals/medical physicians. Machine acquiring knowledge is a subcategory of AI this is particularly involved with the take a look at programs that are improved via records and delight in. The perfecting procedure contains two degrees: instruction and testing out. Machine studying offers a strong tool inside the

medical field to answer numerous fitness issues a good deal speedier. There are two types of machine learning, predictive researching, and autonomous learning. In meticulous study, we form a model with nicely-categorized facts. On the contrary side, an integrated mastery version learns from disparate data. The aim is to develop a first-class computer getting-to-know method this is efficient and accurate for illness prognosis. In this piece, the idea of device getting to know administration is used to are anticipating illnesses. The basic function could be machine learning, wherein we can use a device researching program to be able to properly anticipate diseases at an early stage and enhance impacted person care.

2. PROBLEM DEFINITION

A sophisticated healthcare prediction system is great for minimising healthcare expenditures. Many talents in the health prediction device continue to be useful. Thus we reside at the reducing edge of technology, and yet, if we cannot utilise it appropriately and efficaciously, there may be no use in it. To fix this problem, study is being carried out on healthy forecast frameworks. Some uses or methods use creation. This undertaking demonstrates the merging of both technologies to achieve strong results.

3. OBJECTIVES

3.1 Aim of the Project:

An accurate assessment is reduced while the quality of medical data is limited. In addition, unique areas show special features of certain neighboring diseases, that could impair the forecast of sickness epidemics. But the

contemporary works are taken into account mostly facts-dependent. There are no appropriate methods for managing semi-structured and unorganized data. The suggested computer will care for each based and unstructured data. The precision of the research is extended through the utilization of gadget-studying programs.

3.2 Scope of the Project

The purpose of this artwork is to supply an assistance to support specialists and clients in the hunt and selection of ailments. To achieve this purpose, we create a method that gives the user an inquiry of illnesses that meet the criteria based on the characteristics of the disease, along with the indications and symptoms of the malady, in addition to the gadget of the patient's profile.

4. LITERATURE SURVEY

A machine learning-primarily based approach for personality and price-effective detection of Alzheimer's disease

[Author: Escudero J, Ifeachor E, Zajicek JP, Green C, Shearer J, Pearson S]

- I. When Alzheimer's illness is identified, typically problematic, particularly in the commencement of the diagnostic process inside the phases of moderate cognitive deterioration. However, in this situation, it's kilometers noticeably not probable that treatment can be strong, so a higher diagnosis method would be of great use. We have developed and demonstrated a device learning

- method to customized and price-powerful diagnostics.
- II. It makes use of locally weighted learning to apply a categorization model to each patient and find the set of characters that may be maximally helpful or cost-effective in patient searches. Using ADNI data, we report AD versus controls and MCI patients who improved to AD within one year versus those who did no longer.
 - III. The procedure is analogous to the joint consideration of all warning signs, at the same time considerably reducing the broad variety (and worth) of indicators to obtain a clear forecast for each afflicted individual. Consequently, it may aid to the personalised and successful diagnosis of AD and be advantageous in therapeutic application.

Effect of meteorological conditions on the incidence of hand, foot, and mouth disorder in Wuwei City, North China

[Or: Shan Zheng; Menzhen Wang; Shigun Wang; Kezheng Shang; Lily Hu; Jinrong Dong]

The main objective of this study is to provide a methodological framework for the prevention and prediction of hand, foot and mouth disease dominance, while considering the impact of various climatic conditions on the incidence of hand, foot and mouth disease in Wuwei. The city, the power of China. disease and environmental data for 2008-2010 are collected right here. Techniques of

association assessment, a pair of regression and exponential graphs. The findings verified that 2688 hand, foot, and mouth disease cases had been accumulated from 2008 to 2010, and the typical yearly frequency shifted to 47.Sixty-two/100,000. This illness took place throughout the 12 months inside the city of Wuwei, however, it had a clear peak season. In standard, the instances rose from April and reached their initial height in May, June, and July, accordingly. Another apex takes place in September or October of every 12 months. Various climatic components had amazing influence on the pandemic disease within the four locations, including general temperature, relative humidity, atmospheric stress, rainfall, and evapotranspiration.

Development index for the detection and identity of disorder ranges [Author: Zhang Y, Long JD, Mills JA, Warner JH, Lu W, Paulsen JS]

Spectral data are widely used to gauge the severity of ailments in various plant species. However, such data changed into not being used to predict plant disease levels. This study intended to extend a spectrum disease indicator that could sense wheat leaf rust disorder at exceptional DS degrees. To obtain the aim of the investigation, the absorption spectra of swollen leaves with various fracture symptoms and degrees of DS have been obtained using a spectroradiometer. Pure spectra of several disease symptoms at the leaf scale were then expanded and a new function converted into built-in ones identify the equivalence of delicate levels of illness Sign. The reflection zones with the maximum sensitivity were detected at 675 and 775 nm. Finally, the adjusted

difference between DS and 675/775 was used as a brand new SDI ratio to distinguish between the three disease groups at crown grade. The SDI programme promised to boost the results of diagnosing station diseases in precision plant safety.

Quantitative analysis of coronary heart valve sicknesses according to the known coronary heart technique

[Author: Hu Yuliang; Qiao Junxuan; Wang Haibin; Wei Xiubo]

In order to effectively and effectively assess heart valve illnesses, a new quantitative diagnosis method is suggested to study the four clinical heart valve sounds, especially the characteristic wave of the arterial heart sound. A BIOPAC data collection instrument was used to obtain notifications. Memory facts are communicated to a laptop community via an internet community for storage, analysis and real-time display. A one degree of freedom analysis version is created to identify the feature markers. In addition, diagnostic factors to differentiate between normal and valvular murmurs have been calculated with an easily recognizable visual representation, allowing even an untrained person to easily follow the disease progression. Finally a case look at a coronary individual with a cardiac valve disease before and after surgery process was established, validating the usability and efficiency of the recommended approach.

Nonlinear analysis of coronary heart price variability in sufferers with coronary heart sickness

[Author: Chengyu Liu¹, David Springer², Qiao Li¹, Benjamin Moody³, Ricardo Abad Huang⁴, 5, Francisco J. Chorro⁶]

Using a group of individuals with ischemic heart disease and a control group, the study highlights the therapeutic and predictive consequences of non-linear cardiac variability. controlled organisation of healthy controls of equal age. Three approaches were implemented: Hurst's exponent, irreducible volatility Analysis and estimated entropy. The R-R series Hurst exponent was discovered through application of the rescaling analytic approach. DFA transformed into being used to measure the fractal long-term correlation homes of arterial cardiac charge variability. The estimated entropy of the random variations inside the time gathering is quantified. A quick-time period exposure to fractal magnification became found. In CHD patients, the Hurst index in each exercise test program individually, as well as the estimated entropy, were reduced than within the health management organization.

5. EXISTING SYSTEM

An malady machine can are anticipating, however, can't are expecting disorder subcategories using the frequency of a particular disorder. It cannot anticipate all possible human situations. The contemporary computer best processes based knowledge. The engine of forecasts is vast and unclear. In the past, infinite illness categories have been tried within the taxonomy process. Institutions are

continuously developing a collection of device-learning programs that may correctly identify illnesses. But the exploitation of the main fields is one of a kind. For one component, the prevalent systems are extra steeply-priced, until wealthy people ought to have the funds to spend for such structures. And at the same moment as a manner as males are concerned, even deeply. Second, the reasons for the hypothesis are nonetheless hazy and unsure. Thus, the algorithm can forecast a marvelous condition, but it can not are anticipating the categories of diseases and ailments from the life of a single beetle. For instance, if a group of human beings with diabetes is examined, there's total trust that some of them have a complex chance of arterial illness due to the existence of diabetes. The remainder of the method forecasts all possible tolerance levels.

6.1 DISADVANTAGES OF THE EXISTING SYSTEM

- Does not analyse the illness
- Less privacy
- There is no feedback mechanism

6.2 PROPOSED SYSTEM

The purpose of forecasting many diseases is to apply device learning algorithms and all other tools to construct a machine that forecasts the patient's malady using signs and using taking them, we evaluate them with the machine formerly to be had. Records established By taking those

samples and assessing them for the condition, we make forecasts.

This truth is then categorized as the use of devices to acquire understanding of methods along with logistic regression, linear regression, random forest classifier, and decision tree classifier.

The information then is going into the writer's version in which he examines the hazards connected with the gadget and likewise provides an approximation of the possibility of the device so that it demonstrates varied possibilities, including how the machine performs when the number is "n". Predictions, in addition to recommendations of patients based on their ultimate results, as well as from their indicators, as an instance, what to use and what now not to use can be demonstrated from provided timetables and outcomes. Predicts potential illnesses by analyzing databases including neurological disease, and cardiac problem. To the best of our know-how, no prior study on large-scale scientific analytics has handled both sorts of information.

7. ADVANTAGES OF THE PROPOSED SYSTEM

- Easily analyze the disease
- High Accuracy

7.1 PROPOSED ALGORITHM

- ☐ Support Vector Machine
- ☐ Recurrent Neural Network
- ☐ Principal Component Analysis

8. ARCHITECTURE DIAGRAM OF MACHINE LEARNING

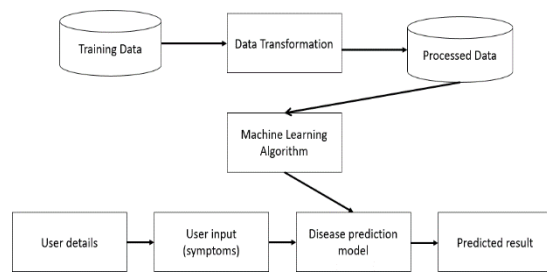


Fig : (i)

8.1 SYSTEM REQUIREMENTS

8.1.1HARDWARE REQUIREMENTS

- Processor: Core i3/i5/i7
- RAM: 2-4GB
- HDD: 500 GB

8.1.2 SOFTWARE REQUIREMENTS

- Platform: Windows Xp/7/8/10
- Coding Language: Python

What is Python?

Python is a popular programming language. It evolved into developed in 1991 through Guido van Rossum.

It is believed:

- web improvement (server);
- application improvement
- arithmetic,
- device creator

9. MODULES

- Data Collection Module
- Preparing the data Module

- Training a model
- Disease prediction Module

9.1.Data Collection Module

Whether basic facts from Excel, Access, textual content papers, etc., this step (accumulating data) provides the groundwork for future research. The higher the variety, quantity, and extend of pertinent records, the higher the device mastery prospects grow to be.

9.2.Preparing the data Module

Exploratory assessment is perhaps one of the methods to specifically investigate the nuances of information, thereby increasing the nutritional content.

9.3.Training a model

This stage involves choosing the ideal collection of guidelines and depicting the model of the record. The cleaned Data is divided into two categories: schooling and employment (the amount changes depending on the circumstances); main element (training statistics) is utilised to broaden the model. The 2nd component is used for comparison.

9.4.Disease prediction Module

The patient is exhibiting indications and symptoms of infection. The device poses a particular query regarding their illness, and the device forecasts the condition mainly based on the signs suggested by means of the afflicted individual.

CONCLUSION

Early identification can help save you and getting the illness. Machine learning creation can assist in the study and discovery of significant causal factors. The

suggested technique creates a machine-studying variant that can anticipate cardiac attacks. The first-rate solution is the SVM collection of principles. The version indicates that new machine-getting-to-know algorithms generally give greater forecast accuracy.

To find effective and precise methods, current tactics are assessed and contrasted. Machine-gaining knowledge of algorithms dramatically increases the accuracy of forecasting the danger of arterial heart and mind tumors, enabling human beings to be identified with an early illness and acquire prophylactic therapy. It can be claimed that machine learning algorithms could have an outstanding ability in terms of forecasting arterial or cardiac heart-related diseases.

This piece provides a top-level perspective of several types of research conducted in this area. The device we recommend is intended toward connecting the space between physicians and patients, on the way to assist each variety of users to acquire their goals. This system offers assistance in forecasting numerous illnesses through the utilization of varied systems acquiring knowledge of algorithms. The cutting-edge approach of optimum structures concentrates most effectively on streamlining this method, which does not assist build a person's self-assurance in the device. By providing Doctor clues in our device, we at the same time guarantee the approval as true of the consumers, so that the Doctor does now not feel that his Business is hurt because of this device.

REFERENCES

- [1] A Narin, C Kaya, and Z. Pamuk, "Automatic detection of coronavirus disease (COVID-19) using x-ray images and deep convolutional neural networks", Mar 2020.
- [2] Rajesh. Ranjan, "Predictions for COVID-19 outbreak in India using Epidemiological models", 2020.
- [3] Mohan Senthilkumar, Chandrasegar Thirumalai, and Gautam Srivastava, "Effective heart disease prediction using hybrid machine learning techniques", *IEEE Access*, vol. 7, pp. 81542-81554, 2019.
- [4] Mohan Senthilkumar, Chandrasegar Thirumalai, and Gautam Srivastava, "Effective heart disease prediction using hybrid machine learning techniques", *IEEE Access*, vol. 7, pp. 81542-81554, 2019.
- [5] M. Bayati, S. Bhaskar and A. Montanari, "Statistical analysis of a low-cost method for multiple disease prediction", *Statistical Methods Med. Res.*, vol. 27, no. 8, pp. 2312-2328, 2018.
- [6] A.K. Shrivastava and S. Kumar Sahu, "Classification of Chronic Kidney Disease using Feature Selection Techniques", *IJCSE*, vol. 6, no. 5, pp. 649- 653, 2018.
- [7] N Chaithra and B Madhu, "Classification Models on Cardiovascular Disease Prediction using Data Mining Techniques", *Journal of Cardiovascular Diseases & Diagnosis*, vol. 6, pp. 1-4, 2018.
- [8] Singh et al., "Heart Disease Prediction Using Machine Learning Algorithms", *2020 International Conference on Electrical and Electronics Engineering (ICE3)*, pp. 452-457, February 2020.
- [9] M. Kavitha; G. Gnaneswar; R. Dinesh; Y. Rohith Sai; R. Sai Suraj "Heart Disease Prediction using Hybrid machine Learning Model", 2020.

- [10] Abhay Kishore “Heart Attack Prediction Using Deep Learning “,2019.
- [11] Escudero J¹, Ifeachor E, Zajicek JP, Green C, Shearer J, Pearson S “Machine learning-based method for personalized and cost-effective detection of Alzheimer's disease”.
- [12] Zhang Y, Long JD, Mills JA, Warner JH, Lu W, Paulsen JS.“Developing an Index for Detection and Identification of Disease Stages”.
- [13] Shan Zheng ; Minzhen Wang ; Shigong Wang ; Kezheng Shang ; Lili Hu ; Jinrong Dong “Effect of Meteorological Conditions on Occurrence of Hand, Foot and Mouth Disease in Wuwei City, Northwestern China”.
- [14] Tamil selvi, M., Karthikeyan, S. “An Ingenious face recognition system based on HRPSM_CNN under unrestrained environmental condition”.
- [15] Jacintha, V., Karthikeyan, S., Sivaprakasam, P., “Surface Flaw Detection of plug valve Material Using Infrared Thermography and Weighted Local Variation Pixel-Based Fuzzy Clustering Technique”.