HEALTH CARE SYSTEM

A Project Report

Submitted By

MIDDE MOUNESH YADAV

210304124073

BANKA DILEEP

210303124240

KOTA AKASH

210303124649

CHITTIBOINA SAI KRISHNA

210303124338

in Partial Fulfilment For the Award of

the Degree of

BACHELOR OF TECHNOLOGY

COMPUTER SCIENCE & ENGINEERING

Under the Guidance of

Prof. GARIMA SHARMA

Assistant Professor

VADODARA

April - 2024

PARUL UNIVERSITY

CERTIFICATE

This is to Certify that Project - 1 (203105499) of 6^{th} Semester entitled "HEALTH CARE SYSTEM" of Group No. PUCSE_322 has been successfully completed by

- MIDDE MOUNESH- 210304124073
- BANKA DILEEP- 210303124240
- KOTA AKASH- 210303124649
- CHITIBOYANA SAI KRISHNA- 210303124338

under my guidance in partial fulfillment of the Bachelor of Technology (B.Tech) in Computer Science & Engineering of Parul University in Academic Year 2023- 2024.

Date of Submission :_____

Prof. GARIMA SHARMA,
Project Guide

Dr. AMIT BARVA,
Head of Department,
CSE, PIET,

Project Coordinator:-
Kruti Sutaria
Parul University.

Acknowledgements

"The single greatest cause of happiness is gratitude."

-Auliq-Ice

Behind our major work which is experienced by every existent in our platoon. During so numerous hurdles and major critical situations this person helped us to reach our thing one step closer and handed a path to reach success. It's veritably inviting and immense pride to work under the guidance of our design companion.Prof. GARIMA SHARMA who saw commodity in us that we didn't see in ourselves. It's the great honor to say that we came more more interpretation of ourselves during your mentorship

MIDDE MOUNESH YADAV - 210304124073

BANKA DILEEP - 210303124240

KOTA AKASH - 210303124649

CHITIBOYANA SAI KRISHNA - 210303124338

CSE, PIET

Parul University,

Vadodara

Abstract

The most crucial element of every person's existence is their health. To maintain good health and frequent monthly checkups are needed. These days, ordinary people do not have much time to get their health checked. In this situation, technology plays a crucial role. Machine Learning techniques are used for a lot of applications. In healthcare, machine learning is crucial in predicting the diseases. It is currently the most popular and successful area of medical treatment. Accurate analysis of medical data aids early disease identification, patient treatment, and community services as a result of machine learning advancements in the biomedical and healthcare sectors. We will create a GUI to ask the user for their symptoms. We are utilizing 4(for example like KNN, DT) machine learning models in this analysis. The output includes the condition, the model's precision, a definition of the disease, and a treatment plan based on the patient's reported symptoms. We are all familiar with the proverb that states, "Early detection and treatment of disease are far preferable to late-stage treatment." To consult the appropriate doctor and maintain good health, this project identifies the illness based on the patient's described symptoms.

Keywords- Machine Learning, GUI, KNN, Decision Tree.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Health is the most important in every human's life. Weekly or monthly check-up of one's health is most important for prevention and also to stay healthy. Nowadays, the individual is not having that much time to go for a health check-up. As we all know the saying which tells that "Prevention of the disease at an early stage is much better than the cure which we take after we get affected by the disease".Accurate and on-time analysis of any health-related problem is important for preventing and treating the illness. The traditional way of diagnosis may not be enough in the case of a serious ailment. Healthcare is the most critical part of human life.Nowadays, so many are not willing to go to a hospital, due to work overload and negligence of their health. The doctors and nurses are putting up a maximum effort to save people's lives without even considering their own lives. There are also some villages that lack medical facilities. We have designed a disease prediction system using an ML algorithm (Naive Bayes, Decision Tree, Random Forest, KNN), find the most accurate algorithm, and used it to find the disease and Tkinter for GUI. After predicting the disease, we will store all the details like patient name, Symptoms they are facing and the disease in Sqlite database. And also, we have created a chatbot using Decision Tree, which will help us in getting accurate predictions by taking into account the symptoms faced by an individual. The Output of the chatbot is the disease, the accuracy of the model, its definition, and the treatment of the particular disease based on the symptoms given by the individuals. This project helps to get an idea about an individual's disease based on the symptoms he/she has, and get the treatment easily by contacting the concerned doctor. A disease predictor can also be called a virtual doctor, which can predict the disease based on symptoms. This disease predictor system can be most useful as it identifies the disease without even contacting the individual.

1.1 Overview

A disease is a condition that affects the individual functioning of the body totally. Diseases if ignored will result in the death of a human. Diseases can be identified by the symptoms of the body of an individual. Health is the most important in every human's life. Weekly or monthly check-up of one's health is most important for prevention and also to stay healthy. Nowadays, the individual is not having that much time to go for health check-ups. In this situation, where everything has turned virtual if the treatment process can be completed using an automated program it would be really helpful for the patients. As we all know the saying which tell us that "Prevention of the disease at an early stage is much better than the cure which we take after we get affected by the disease". Healthcare is the most crucial part of human life. Nowadays, so many are not willing to go to a hospital, due to work overload and negligence of their health. The doctors and nurses are putting up a maximum effort to save people's lives without even considering their own lives. There are also some villages that lack medical facilities.

Accurate and on-time analysis of any health-related problem is important for preventing and treating the illness. The traditional method of diagnosis may not be sufficient in case of a serious illness. In this case, everything is virtual but doctors and nurses are doing all kinds of efforts to save one's life which incidentally is quite risky for their own life. There are also some remote villages which lack full hospital facilities or doctors. The same dataset was fed into the ML models like Naive Bayes, Random Forest, KNN, and Decision Tree. While processing the information, symptoms are taken as inputs and the disease was received as an output.This project helps to get an idea about an individual's disease based on the symptoms he/she has, and get the treatment easily by contacting the concerned doctor. We can store all the details like symptoms, diseases etc for future reference

1.2 Machine Learning

Machine learning could be a subfield of computer science (AI). The goal of machine learning typically is to know the structure information of knowledge and match that data into models which will be understood and used by folks.Though machine learning could be a field inside technology, it differs from ancient process approaches.

In ancient computing algorithms are sets of expressly programmed directions employed by computers to calculate or downside solve. Instead, computer algorithms give computers the ability to learn from knowledge inputs and use applied math analysis so as to output values that fall inside

a special variety. thanks to this, machine learning helps computers create models from sample knowledge in order to alter decision-making processes based on knowledge inputs.

1.3 Machine Learning Strategies

Techniques In machine learning, the tasks are commonly categorized into wide classes. These classes are based on how learning takes place or how, the feedback from the education is delivered to the system developed. The two most widely used machine learning strategies are supervised learning that trains algorithms with example input and output information that are tagged with the help of humans, and unsupervised learning that provides the algorithmic program without such tagged information to let it find structure within its file.

1.4 Supervised Learning

In supervised learning, the pc is fed example inputs that square measure labelled with their wanted outputs. The objective of this technique is for the algorithmic program to be ready to ”learn” by comparing its actual output with the ”taught” outputs to search out errors, and modify the model consequently. Supervised learning thus uses patterns to predict label values on extra unlabelled information. For example, in supervised learning, the algorithm can be trained on input data consisting of images of sharks which would be tagged as fish and images of oceans that may be tagged as water. Then, the algorithm should, therefore, be able to identify untagged images of a shark as a fish and determine an untaged image of an ocean as water. A common application of supervised learning is to take historical data and use that to predict statistically probably future events. It will use historical stock exchange info to anticipate approaching fluctuations or be used to filter spam emails. In supervised learning, usually, the input files will be labeled photos of dogs to classify unlabeled photos of dogs

1.5 Unattended Learning

Because in unattended learning information is unlabeled, and the learning rule is left to find patterns that are common among its input file. The objective of unattended learning is also as simple as discovering hidden patterns at intervals within a dataset, though it should also have an objective of feature learning, which allows the procedure machine to automatically discover the representations that are needed to classify data. Unsupervised learning is essentially applied to transactional information. You will end up with an over-sized dataset of customers and their purchasing behaviour however, as a human being, you will most likely not be able to sum up

what corresponding features will be extracted from customer profiles and their patterns of buying. With this knowledge input to the Associate in Nursing unattended learning algorithm, it should be concluded that women of a certain age group who come to purchase unscented soaps most likely be pregnant, and hence, an advertisement campaign associated with pregnancy and baby will be marketed.

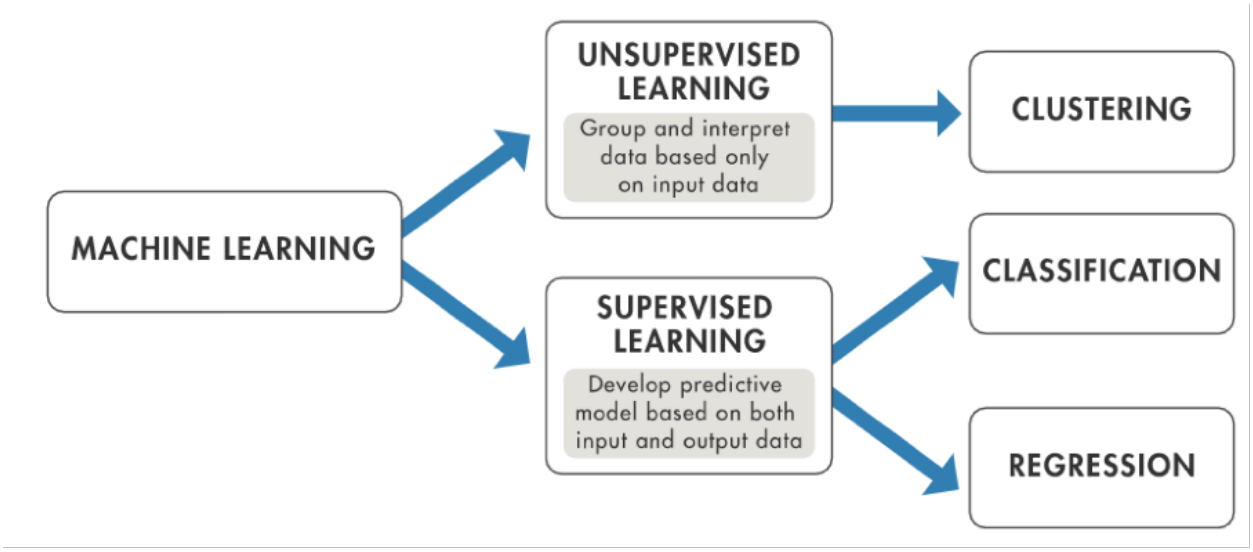

Figure 1.1: Machine Learning Classification

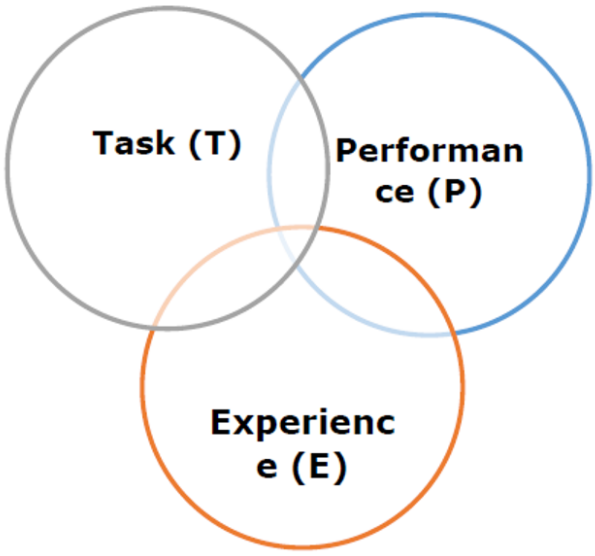

Figure 1.2: Machine Learning Task

Chapter 2

Literature Survey

2.1 Related work

1. Tarigoppula V.S Sriram et al. in [1] collects the voice dataset from UCI Machine Learning repository and train four algorithms on that dataset. The result is the prediction of Parkinson Disease by considering the most accurate algorithm.

2. Shubham Bind et al. in [2] studies about all the available researches in literature to predict the Parkinson diseases.

3. K. Gomathi, D. Shanmuga Priya in [3] used different data mining techniques to predict Heart disease, Breast Cancer, Diabetes. The models used are Decision Tree and Naive Bayes Classifier. Performance of both the models was compared and the best classifier is used to predict the above diseases.

4. Isha Pandya et al. in [4] used two supervised machine learning algorithms Decision Tree, accuracy 91perccent and Naïve Bayes classifier, accuracy 87percent. Here, they used the combination of both to get the best accuracy. Naïve Bayes Classifier accuracy should be improved.

5. Akash C. Jamgade, Prof. S. D. Zade in [5] paper determined the most danger diseases which occur in a person in a locality and community. But, the data collection is difficult.

6. Siddhika Arunachalam in [6] six classification algorithms are used after analyzing 14 attributes in the dataset. But, we may get confused which algorithm to use.

7. Ionela-Catalina ZAMFIR, Ana-Maria Mihaela IORDACHE methodologies in [7] used are Support Vector Machines, Artificial Neural Networks, K-Means Algorithm, Decision Trees,

Logistic Regression and predicted diseases are breast cancer, lung cancer, heart diseases, diabetes, thyroid or kidney diseases.

8. H BENJAMIN FREDRICK DAVID in [8] predicted the occurrence of heart disease using ensemble learning algorithms. But, the Research work can be made to produce an impact in the accuracy of the Decision Tree and Bayesian Classification.

9. Harshit Anand et al. in [9], domains of Machine learning and Data Science are used and models are built using numpy, pandas, sklearn, and so on and the model are deployed using Django.

10. Goutam Chakraboty et al. in [10], takes into account six features from 23 features in the dataset and predict the risk of chronic kidney disease. It is used to reduce the impact of Chronic Kidney disease (CKD), where creatinine test is not available for all.

11. Durga Praveen et al. in [11], applied 5 models namely KNN, SVM, Random Forest, Naïve Bayes and Adaboost and found that KNN, Adaboost has the highest accuracy of all the models. So, any of these two are used for prediction and prevention of the liver disease.

12. Sejin Park et al. in [12], early prediction of disease using the previous real-time stroke symptoms. It is implemented at a low cost. Random Forest algorithm is used for validating clinical significance.

13. Ahan Chatterjee et al. in [13], they have used machine learning models like Decision tree, SVM, Random Forest, and so on, find the best classifier using the accuracy and use it to predict cancer disease risk in the early stage and prevent it. Simulation model is also designed to manage the patient flow in OPDs.

14. Sergio Grueso et al. in [14], they have used dataset taken from ADNI database and selected 47 out of 159 studies for analysis. Deep learning combined with multimodal and multidimensional data is used to achieve the best performance.

15. Upendra Kumar in [15], have used Computer-Aided Pre-Screening Tool (CAPST) which improves the accuracy of diagnosis in medicine. By this, fast and accurate prediction risk of disease is found.

2.2 Open Problems in Existing System

From the above literature survey, we have inferred that Many of the existing machine learning models for health care analysis are concentrating on one disease per analysis. One analysis is for diabetes analysis, one for cancer analysis, and one for skin diseases like that by implementing various algorithms on particular datasets.

After implementing various algorithms, the most accurate one is selected and it is used for the prediction of disease. Sometimes, we may get confused about what algorithm to use. Also, all the systems find only the particular disease and not the disease based on the symptoms.

Diseases may be predicted by a machine, but subtypes of diseases brought on by the presence of one disease cannot. It is unable to account for all potential human situations. Only structured data is handled by the current system. The prediction system is vague and unclear. Several illnesses estimate classes have been developed and are now being used. The established organizations set up a combination of machine learning algorithms that are carefully precise in predicting diseases.

The best algorithm is chosen after being tested against a variety of others, and it is then used to disease prediction. Choosing the right algorithm might be confusing at times. Additionally, none of the methods identify an illness based solely on its symptoms; instead, they all identify a specific condition.

2.3 Disadvantages of The Existing System

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

Chapter 3

Analysis / Software Requirements Specification (SRS)

3.1 Proposed System

We are proposing a system, which uses a Tkinter for the GUI interface. It is a simple user interface and is also time efficient. Our aim with is to get the disease based on symptoms given by the user. The domain we will use is machine learning, in that we will be using Naïve Bayes Classifier, Random Forest Classifier, KNN, Decision Tree, which will help us in getting the most accurate predictions easily, and also the accuracy is given as output. To reduce time consumption, we will ask only fewer questions namely the name of the individual and the symptoms the individual is facing. In this way, our system will be less time-consuming and give accurate predictions., which will help us in getting accurate predictions by taking into account the symptoms faced by an individual. The suggested method of multiple illness prediction by machine learning comprises of a system that predicts the disease of the patient based on their symptoms, and by using those symptoms, we compare them to the dataset of the system that was previously available. By comparing those datasets to the patient's disease, we can precisely determine the patient's disease proportion. The system's prediction model receives the dataset and symptoms and pre-processes the data for subsequent use. After inputting or selecting the various symptoms, the user selects the features. We will also compare different algorithms to see which one produces results most quickly and effectively. We will only inquire about the person's name and any symptoms they may be experiencing to save time. This will make our algorithm faster and more accurate in making predictions. Additionally, we are leveraging Decision Tree to build a chatbot that will enable us to make precise predictions by accounting for a person's symptoms.

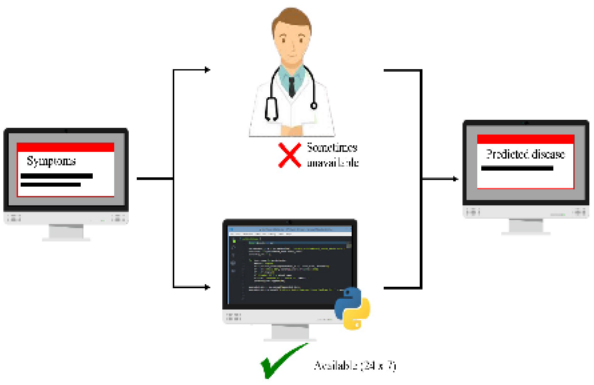

Figure 3.1: Proposed approach for predicting illness. It's possible that the doctor won't always be on call. We can use this method at any moment to predict the disease depending on our symptoms.

Algorithm

- Import libraries and Dataset.
- Data Pre-processing
- Data Visualization
- Model Building
- Model Evaluation
- Deployment of the model
    1. GUI
    2. Chatbot
- Based on user-provided symptoms, predict the disease.

ADVANTAGES OF THE PROPOSED SYSTEM

- Easily analyze the disease
- High Accuracy

3.2 Programming Language

3.2.1 PYTHON

Python is the best programming language fitted for Machine Learning. In step with studies and surveys, Python is the fifth most significant language yet because the preferred language for machine learning and information science. It's owing to the subsequent strengths that Python has:

- **Easy to be told and perceive:** The syntax of Python is simpler; thence it's comparatively straightforward, even for beginners conjointly, to be told and perceive the language.
- **Multi-purpose language:** Python could be a multi-purpose programing language as a result it supports structured programming and object-oriented programming yet as practical programming.
- **Support of open supply community:** As an open supply programming language, Python is supported by an awfully giant developer community. Because of this, the bugs square measure simply mounted by the Python community. This characteristic makes Python terribly strong and adaptative.

HISTORY OF PYTHON

- Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.
- Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.
- Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).
- Python is maintained by a core development team at the institute, although Guido van Rossum still plays a vital role in directing its progress.

APPLICATION OF PYTHON

- GUI Programming: Python allows for GUI applications that can be developed and can be easily ported to many system calls, libraries, and windows systems like the Windows MFC, Macintosh, and the X Window system of Unix.
- Scalable: Python has a much better structure and support for large programs than shell scripting.
- Large standard libraries to solve common tasks: It has a number of standard libraries which makes life of a programe.
- Easy-to-learn Python has fewer keywords, a simple structure, and a very clearly defined syntax. This allows the student to pick up the language quickly.

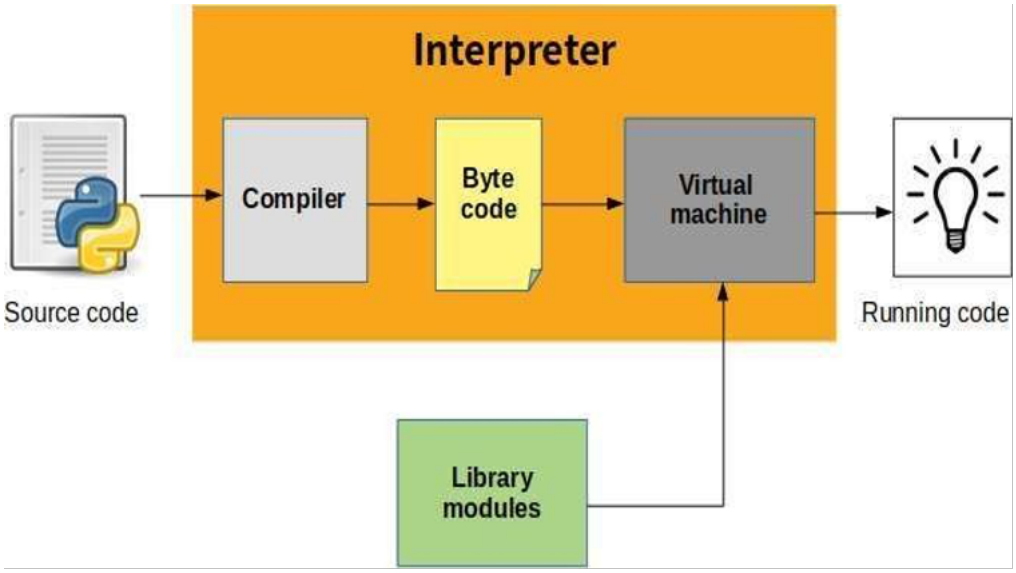

Figure 3.2: Execution of source code

- Easy-to-read Python is more visible to the eyes and also better defined.
- Easy-to-maintain Python's source code is fairly easy-to-maintain.
- Standard Library A very broad standard library Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.

  Mode Python has support for an interactive mode which allows interactive testing and debugging of snippets of code. Features of Python

Features of Python

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can easily be integrated with C, C++, COM, ActiveX, CORBA, and Java.

3.2.2 Domain

Machine learning can be a subfield of computer science (AI). In other words, the final objective of machine learning is to comprehend information knowledge structure, so it can fit in models that will be understood and used by humans. This field of machine learning is a sub-area in technology but

distinguishes itself from a conventional process approach. Algorithms in traditional computing are series of explicitly programmed instructions used by computers when they try to solve a problem or compute something. Machine learning algorithms instead give computers to coach on knowledge inputs and use applied math analysis so as to output values that fall inside a particular vary. Thanks to this, machine learning facilitates computers in building models from sample knowledge so as to modify decision-making processes supported by knowledge inputs.

Chapter 4

System Design

4.1 System Architechure

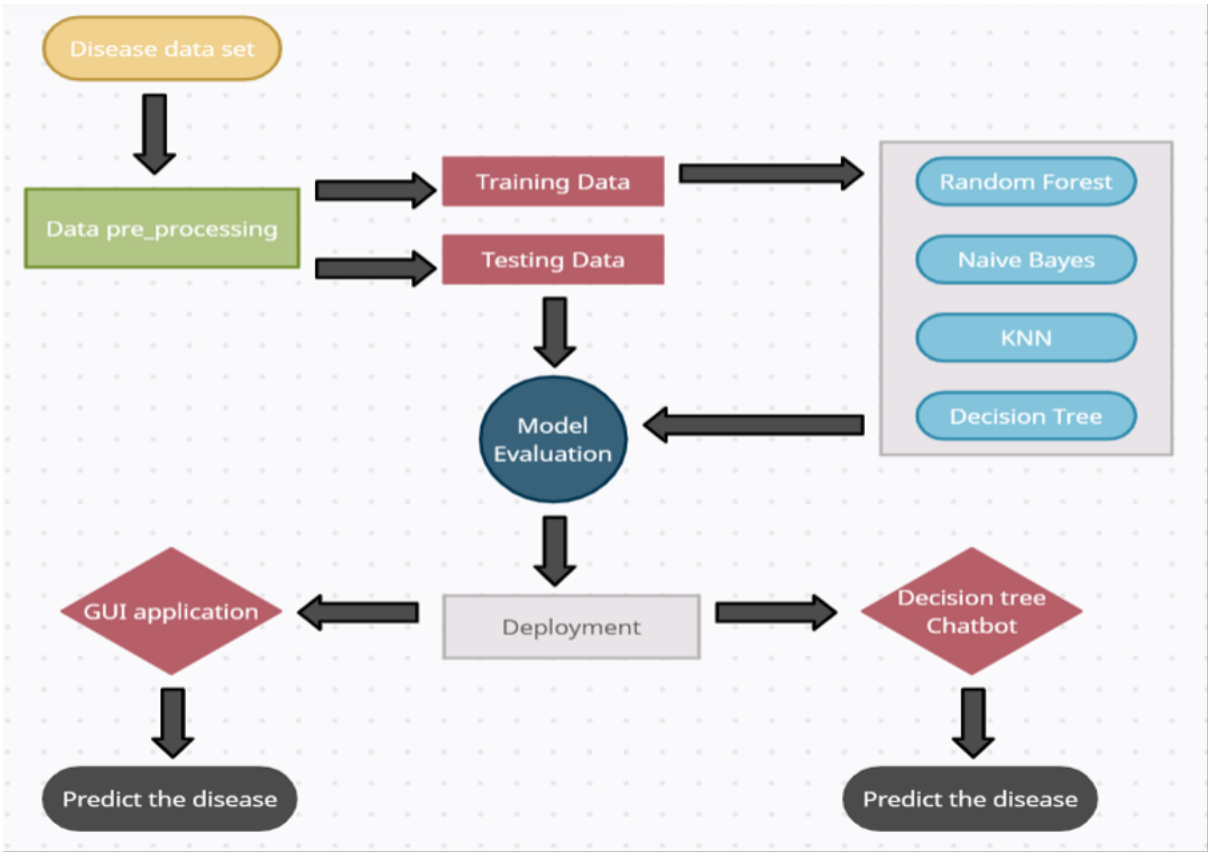

Figure 4.1: System Architecture

4.2 Alogrithms

4.2.1 Naive Bayes

Naive Bayes algorithm is a supervised learning algorithm, based on Bayes theorem which is used for solving classification problems. It is mainly used in text classification that includes a high-

dimensional training dataset. Na¨ıve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions. It's a probabilistic classifier, meaning it predicts on the basis of probability of an object. Some of the well-known examples for Naive Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles. The Na¨ıve Bayes algorithm consists of two words Na¨ıve and Bayes, which can be defined as:

- Naive: It is called Naive because it assumes that the occurrence of a certain feature is independent of the occurrence of other features. Such as if the fruit is identified on the bases of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple. Hence each feature individually contributes to identify that it is an apple without depending on each other.

- Bayes:It is called Bayes because it depends on the principle of Bayes' Theorem Naive Bayes is called naive because it assumes that each input variable is independent. This is a strong assumption and unrealistic for real data; however, the technique is very effective on a large range of complex problems.

Naive Bayes is a classification algorithm, which is suitable for the purposes of binary and multiclass classification. Na¨ıve Bayes works far better with categorical input variables than with numerical variables. It is useful for making predictions and forecasting data based on historical results. It is a machine learning algorithm that is used for a classification problem based on Bayes theorem. Mainly this is used for doing text classification. Bayes theorem can be defined as: P (C—X) = P (X—C) . P (C) P (X) P(C—X) is the probability of hypothesis C for the given data X. This is called the posterior probability. P(X—C) is the probability of data X given that hypothesis C was true. P(C) is the probability of hypothesis C being true. This is called the prior probability of C. P(X) is the probability of the data and evidence of data and is called marginal probability. Naive Bayes Classifier uses the Bayes' theorem to predict membership probabilities for each class such as the probability that given record or datapoint belongs to a particular class. The class with the highest probability is considered as the most likely class. This is also known as the Maximum A Posteriori (MAP). Out of the 5 different Naive Bayes classifiers under sklearn. naive bayes, the 3 most widely used ones are Gaussian, Multinomial, and Bernoulli.

Advantages of Naive Bayes Classifier:

- Naive Bayes is one of the fast and easy ML algorithms to predict a class of datasets.

- It can be used for Binary as well as Multi-class Classifications.
- It performs well in ulticlass predictions as compared to the other Algorithms.
- It is the most popular choice for text classification problems.

Disadvantages of Naive Bayes Classifier:

- Naive Bayes assumes that all features are independent or unrelated, so it cannot learn the relationship between features.

Applications of Naive Bayes Classifier:

- It is used for Credit Scoring.
- It is used in medical data classification.
- It can be used in real-time predictions because Naïve Bayes Classifier is an eager learner.
- It is used in Text classification such as Spam filtering and Sentiment analysis.

Types of Naive Bayes Model: There are three types of Naive Bayes Model, which are given below:

- **Gaussian:** The Gaussian model assumes that features follow a normal distribution. This implies if predictors are continuous rather than discrete, then the model assumes that these are samples from the Gaussian distribution.
- **Multinomial:** The Multinomial Na¨ıve Bayes Classifier is used when data is multinomial distributed. It is mainly used for document classification problems; it means a specific document falls under which category like sports, politics, education, etc. The, classifier uses the frequency of words for the predictors.
- **Bernoulli:** The Bernoulli classifier works similar to the Multinomial classifier but the predictor variables are the independent Booleans variables. Such as if a particular word is present or not in a document. This model is also famous for document classification tasks.

Gaussian Naive Bayes It follows the same procedure as the Naive Bayes. For Naive Bayes, we require a dataset to be categorical and in the case of Gaussian Naive Bayes, we need all feature attributes in the dataset to be continuous. GNB is a classification method applied in ML based on the probabilistic approach and Gaussian distribution. Gaussian Naive Bayes assumes that each

parameter, also referred to as features or predictors, has an independent capacity in predicting the output variable.Naive Bayes is a generative model. (Gaussian) Naive Bayes assume each class follows a Gaussian distribution. The difference between QDA and (Gaussian) Naive Bayes is that Naive Bayes assumes independence of the features, which means the covariance matrices are diagonal matrices

4.2.2 Random Forest

Random Forest is a supervised learning, used for both classification and Regression. It is a ensemble learning model that operates by constructing a multitude of decision trees at training time. The logic behind the random forest is that it uses the bagging technique to create random sample features. The only difference between the decision tree and the random forest is that the process of finding the root node and splitting the feature node will run randomly. The steps are provided below:

- Load data where it consists number of features representing behavior of the dataset.
- The name given to the training algorithm of random forest is bootstrap algorithm or bagging technique to select n feature randomly from m features, that is to say to create some random samples, this model trains a new sample to out of bag sample (1/3 rd of the data) used to calculate unbiased OOB error.
- Compute the node d, using the best split. Break the node into sub-nodes.
- Repeat the above steps, to get n number of trees.Calculate how many votes each tree had made for a predicting target. Class-wise, which got the maximum votes, is the final prediction of the random forest.

We have to know about the nature of the model before building it. There are parameters referred to as hyperparameters in the Random forest classifier ML algorithm that might be tuned to attain maximum performance. The applied hyperparameters during the model modeling are:

- **n estimators:**Number of decision trees within the forest(int, default = 100).
- **bootstrap:**Whether bootstrap samples have been used when building trees. For false, the whole dataset gets used for the construction of each tree(bool, default = True).
- **max depth:**Absolute maximum depth of the forest. If None, then in full, nodes are expanded until all leaves are pure or until all leaves contain less than min samples split samples. The default is that the tree grows to full-depth. (int, default=None).

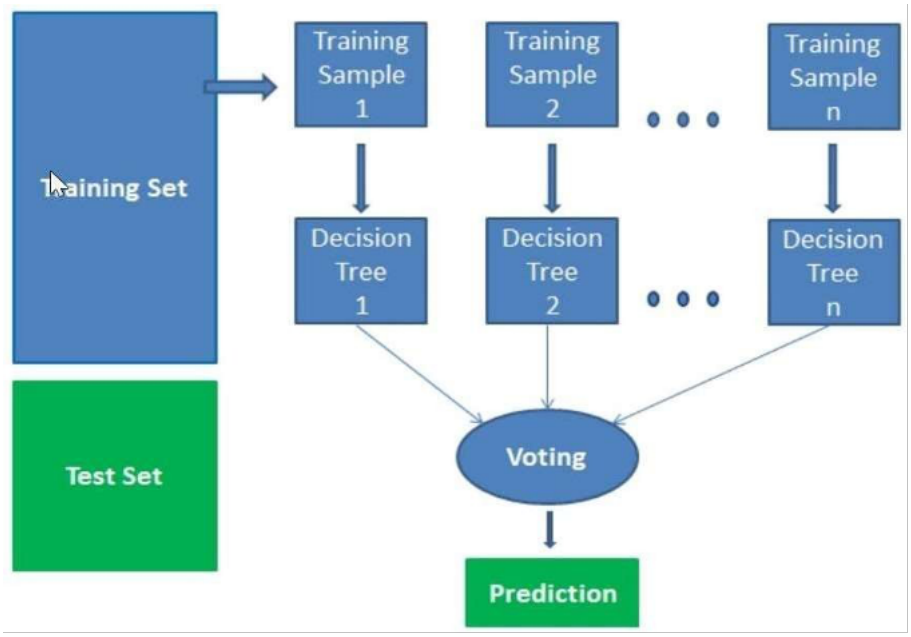

Figure 4.2: Working model of Random Forest algorithm

- **min samples leaf:**Minimum number of samples needed to reach a leaf node. At any depth a split point will only be evaluated if it leaves at least min samples leaf training examples in each of the left and right branches. This might have the effect of smoothing the model, especially in regression. If int, then considers min samples leaf as the minimum. If float, then 'min samples leaf' is a fraction, an ceil(min samples leaf * n samples) is the minimum number of samples for each node. (int or float,default=1).

- **random state:** Controls both the randomness of the bootstrapping of the samples used when building trees and the sampling of the features to consider when looking for the best split at each node(int, default=None).

Here the n estimators, bootstrap and random state are constant n estimator represents no. of decision trees, random state is given to test values for same shuffle data.

Classification in random forests Classification in random forests employs an ensemble methodology to attain the outcome. The training data is fed to train different types of decision trees. This dataset contains observations and attributes, which will be randomly chosen at the time of splitting of nodes.

A rain forest system is a collection of multiple decision trees. Each decision tree consists of decision nodes, leaf nodes, and a root node. The leaf node of each tree is the final output that comes out of that specific decision tree. The final output is selected with a majority-voting system. Output of the selected by most of the decision trees in this case.

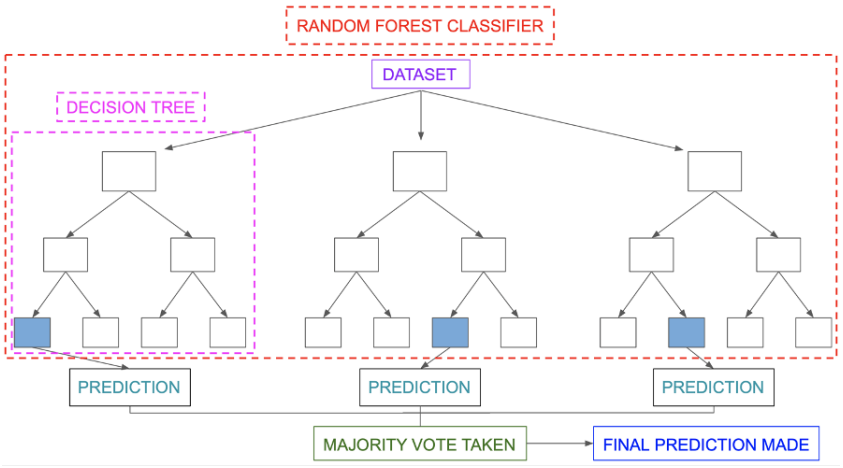

Figure 4.3: Working of Random Forest Classifier

Example: Suppose a dataset is there which contains several images of fruits. So, this dataset is provided it then distributes the subsets to multiple decision trees and sends them to the Random forest classifier. Each of the decision trees performs a prediction result in the training phase, and a new data then the classifier by Random Forest makes a decision using most of its results.A decision. Consider the photograph below.

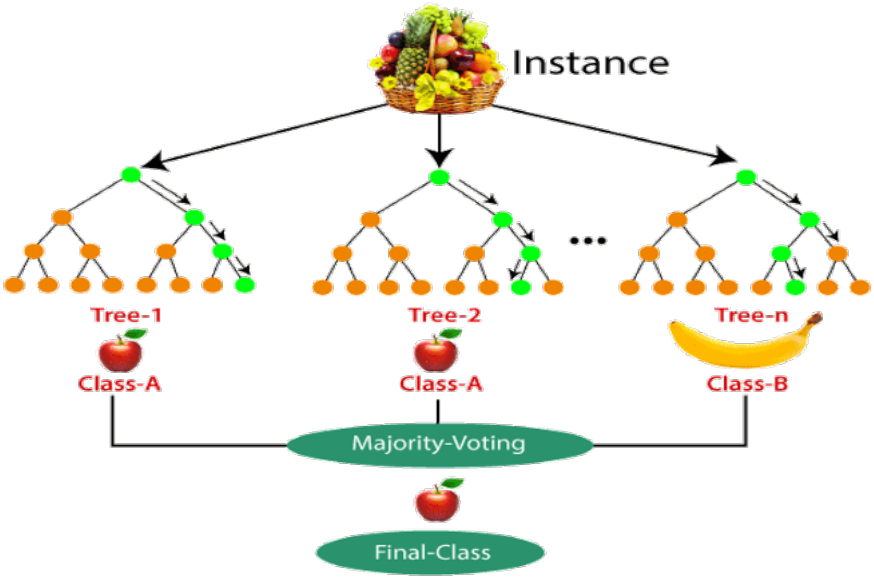

Figure 4.4: Random Forest Classifier with example

Features of a Random Forest Algorithm

- It's more accurate than the decision tree algorithm.
- It provides an effective way of handling missing data.
- It can produce a reasonable prediction without hyper-parameter tuning.
- It solves the issue of overfitting in decision trees.

- In every random forest tree, a subset of features is selected randomly at the node's splitting point.

APPLICATIONS OF RANDOM FOREST

There are mainly four sectors where Randomforest mostly used:

1. Banking: Banking sector mostly uses this algorithm for the identification of loan risk.
2. Medicine: With the help of this algorithm, disease trends and risks of the disease can be identified.
3. Land Use: We can identify the areas of similar land use by this algorithm.
4. Marketing: Marketing trends can be identified using this algorithm.

Advantages of Random Forest

Random Forest is capable of performing both Classification and Regression tasks.

- It is capable of handling large datasets with high dimensionality.
- It enhances the accuracy of the model and prevents the overfitting issue.

Disadvantages of Random Forest

- Although random forest can be used for both classification and regression tasks, it is not more suitable for Regression tasks.

4.2.3 Decision Tree

Decision Tree is a Supervised learning algorithm, which can be used for both classification and Regression problems, though it is especially preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules, and each leaf node represents the outcome. In a Decision tree, there are two types of nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas the Leaf nodes are the output of that decision and do not have a single further branch. A Decision Tree is an algorithm, whose input and output or known. Information gets divided repeatedly using a particular parameter. The decision nodes specify the decision at which parameter should be spilled. These are the output bought by the decisions. A decision tree asks for either true or false to divide the data. Decisions or the test is made by features of the given dataset. It is a graphical presentation for getting all the possible

solutions to a problem/decision based on given conditions. It is called a decision tree because it is just like any other tree, but it starts from the root node that further expands on the other branches and constructs a tree-like form. In order to create a tree, we use the CART algorithm, which stands for the Classification and Regression Tree algorithm. A decision tree just asks a question, and splits the tree into subtrees depending upon the answer to the question in the form of Yes/No. The general structure of a decision tree has been explained in the following diagram: There are many algorithms present in Machine Learning, so always remember the key point that while developing a machine learning model, the best algorithm must be selected according to the given dataset as well as according to the problem. Now, let's see the two reasons for using the Decision tree:

- A decision tree generally resembles human thinking ability while deciding, so it is easy to understand.
- A decision tree generally resembles human thinking ability while deciding, so it is easy to understand.

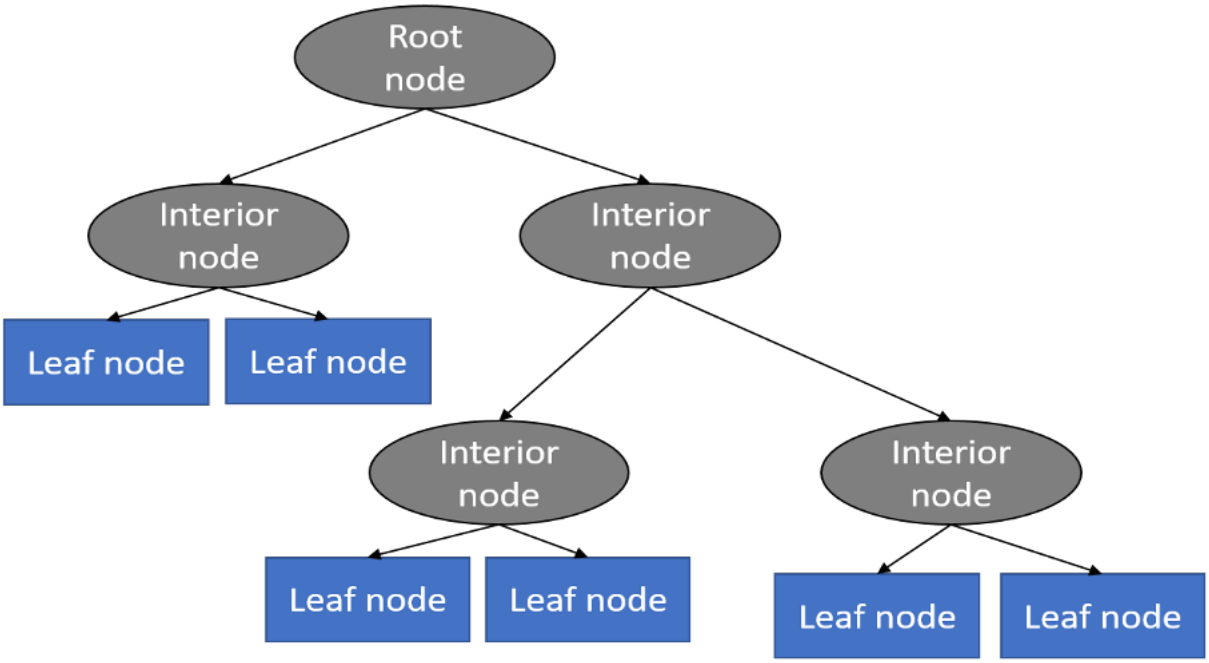

Figure 4.5: Decision Tree

Root Node: Root node is from where the decision tree starts.It represents the entire dataset, which further gets divided into two or more homogeneous sets.

Leaf Node: Leaf nodes are the final output node, and the tree cannot be segregated further after getting a leaf node.

Splitting: Splitting is the process of dividing the decision node/root node into sub-nodes according to the given conditions.

Branch/Sub Tree: A tree formed by splitting the tree.

Pruning: Pruning is the process of removing the unwanted branches from the tree.

Parent/Child node: The root node of the tree is called the parent node, and other nodes are called the child nodes.

Decision tree working:

- Step-1: Begin the tree with the root node, says S, which contains the complete dataset.
- Step-2: Find the best attribute in the dataset using Attribute Selection Measure (ASM).
- Step-3: Divide the S into subsets that contains possible values for the best attributes.
- Step-4: Generate the decision tree node, which contains the best attribute.
- Step-5: Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.

Advantages of the Decision Tree:

- It is simple to understand as it follows the same process which a human follow while making any decision in real-life.
- can be very useful for solving decision-related problems.
- helps to think about all the possible outcomes for a problem.
- is less requirement of data cleaning compared to other algorithms.

Disadvantages of the Decision Tree:

- decision tree contains lots of layers, which makes it complex.
- may have an overfitting issue, which can be resolved using the Random Forest algorithm.
- more class labels, the computational complexity of the decision tree may increase.

4.2.4 K-Nearest Neighbors

K-Nearest Neighbour is one of the simplest algorithms of Machine Learning, which uses the Supervised Learning technique. The assumption of K-NN algorithm of similarity between the new case/data and existing cases makes the new case fit into the most similar category of the available categories. The K-NN algorithm keeps storing all the available data and classifies a new data point

Figure 4.6: K-Nearest Neighbour

based on its similarity. That is, when new data arises then it can be classified very easily in to a well suite category by using K-NN algorithm. K-NN algorithm may be applied for the Regression as well as for the classification but it is mainly used for Classification problems. K-NN is a non-parametric algorithm that is used without any assumption regarding the underlying data. This algorithm is also known as lazy learner because it does not learn on the training set itself immediately. It stores the dataset, and at the time of classification, it acts upon the dataset. The KNN algorithm in the training phase only stores the dataset, and then it simply classifies the data into a category which is much more similar to new data when it gets new data. It's the most popular supervised machine learning method that is pretty easy to apply for imputation of missing values. This method applies to both classification and regression. This is based on the principle that among the observations that are most "similar" to a particular point in any data set, the ones that are nearest to it will be those most "similar". Using this fact we are able to classify a new, unforeseen point by virtue of the properties of the nearest established sites. To get the number of nearby observations in order to use it in the method the user may choose K.

Example: Let's say we have a picture of a species that resembles both cats and dogs, but we aren't sure if it is one or the other. Therefore, since the KNN algorithm is based on a similarity metric, we can utilise it for this identification. Our KNN model will look for similarities between the new data set's features and those in the photos of cats and dogs, and based on those similarities, it will classify the new data set as either cat- or dog-related.

The K-NN working can be explained on the basis of the below algorithm:

- Step-1: Select the number K of the neighbors

Figure 4.7: graphical view of knn algorithm classification

- Step-2: Calculate the Euclidean distance of K number of neighbors
- Step-3: Take the K nearest neighbors as per the calculated Euclidean distance.
- Step-4: Among these k neighbors, count the number of the data points in each category.
- Step-5: Assign the new data points to that category for which the number of the neighbor is maximum.
- Step-6: Our model is ready.

Suppose we have a new data point and we need to put it in the required category.

- Firstly, we will choose the number of neighbors, so we will choose the k=5.
- Next, we will calculate the Euclidean distance between the data points. If (x1, y1) and (x2, y2) are the two points in the two-dimensional plane, the Euclidean distance formula is given by:

$$Euclideandistance, d = \sqrt{x_2 - x_1^2 + y_2 - y_1^2} \tag{4.1}$$

- By calculating the Euclidean distance, we got the nearest neighbors, as three nearest neighbors in category A and two nearest neighbors in category B.
- As we can see the 3 nearest neighbors are from category A, hence this new data point must belong to category A.

Below are some points to remember while selecting the value of K in the K-NN algorithm:

- There is no particular way to determine the best value for ”K”, so we need to try some values to find the best out of them. The most preferred value for K is 5.
- A very low value for K such as K=1 or K=2, can be noisy and lead to the effects of outliers in the model.
- Large values for K are good, but it may find some difficulties.

Advantages of KNN Algorithm:

- It is simple to implement.
- It can be more effective if the training data is large.
- It is robust to the noisy training data.

Disadvantages of KNN Algorithm:

- Always needs to determine the value of K which may be complex some time.
- The computation cost is high because of calculating the distance between the data points for all the training samples.

The project contains three parts:

- DATASET COLLECTION.
- TRAIN AND TEST THE MODEL.
- DEPLOY THE MODELS.

Dataset Collection- We collected datasets from Kaggle notebooks. The dataset is of two types one is training dataset and the another one is testing dataset. Training dataset is used for training our model and with the help of testing dataset we will find how our model performed in terms of accuracy score, confusion matrix etc. The dataset contains the symptoms and the corresponding disease. It contains 4920 rows and 133 columns. In our dataset there are 133 symptoms and 41 diseases related information. So, by using our model we can predict nearly 41 diseases just by giving the symptoms they are facing.

We followed basic machine learning steps first we imported all the necessary libraries next we loaded our training and testing dataset with the help of pandas library. Next, we generated a report of our dataset by using pandas profiling wherein we get to know the each column size,

datatype and missing values, null values, correlation etc. basic data exploration will be done . we get to know the basic details of the dataset its mathematical and analytical values. Next data visualization will be done here we created a scatter plot and bar graph for both training and testing datasets. In data preprocessing we will replace the output class label with numbers ranging from 0 to 40 because it is a classification problem entire dataset will be in numbers rather than strings or other datatypes.

Train and test the model- We used Naïve Bayes Classifier, Random Forest Classifier, KNN, Decision tree Classifier as a model to train the dataset. After training, we tested the model and found its metrices. we will train our model with the training dataset with four machine learning algorithms. We will create a GUI application using python library called tkinter using TK toolkit. Tkinter is a python package wherein we can create login pages, small web application which are easy to use for the users. While training we created a GUI application so that when user enters their input our model predict 4 outputs for each input given by the user. Four outputs each algorithm predict one output so in that way we get four outputs. We will also evaluate our model with different classification metrices like accuracy score, confusion matrix etc.

Deploy the models- Deployed our model by creating a Graphical User Interface to get the name, and symptoms of an individual. By this, we will get the disease and accuracy of the model as the output. which helps an individual to get the corresponding disease by checking whether he/she is being faced with the symptoms. Also, we are storing the details in SQLite database. When a patient used the GUI application by entering his name , symptoms our model will predict the diseases by 4 algorithms and their accuracies. These all details like name, symptoms and the predicted disease will be stored so that they can refer easily. We also get to know the confusion matrix, scatterplot of the disease etc.

We have also created a chatbot using Decision Tree which helps an individual to get the corresponding disease by checking whether he/she is being faced by the symptoms. In this model our output includes the condition, the model's precision, a definition of the disease, and a treatment plan based on the patient's reported symptoms. In this way machine learning when implemented in healthcare can help in satisfying the individual and also take care of their particular disease easily. GUI application which is the easiest model helps to get the idea about the disease of an individual based on the symptoms he/she have, and get the treatment easily by contacting the concern doctor.

Following are the steps to do this project (use Jupyter Notebook):

1. Collect the dataset.
2. Import the necessary libraries.
3. Visualize the dataset.
4. Train the dataset using the Naïve Bayes classifier, KNN, Random Forest Classifier and Decision Tree Classifier.
5. Test the model and find the accuracies of four algorithms.
6. Deploy the model:
    - GUI Interface using Tkinter
    - Chatbot using Decision Tree
7. Predict the disease based on the symptoms given by the user.

Chapter 5

Methodology

5.1 Aim And Scope of The Project

The main aim of the project is to predict the diseases from the symptoms given by the users. Our model will process with four machine learning algorithms like Naïve Bayes, Random Forest, KNN and Decision Tree. We can predict nearly 42 diseases with our model. Many existing models predict only one disease but we can predict 42 diseases. Our models can solve so many problems. Our model is very scalable when compare to other models. The Scope of the project is it will solve major problems in health care domain by predicting the diseases at an early stage.

5.2 Objective of The Project

Developing a project based on machine learning (ML) algorithms for the prediction of any disease can help in a more accurate diagnosis than the conventional method is the main objective of the project. We have designed a disease prediction system using an Machine Learning algorithms, Based on the symptoms of an individual, the Machine learning model gives the output, i.e., the disease that the individual might be suffering from. This project helps to get an idea about an individual's disease based on the symptoms he/she has, and get the treatment easily by contacting the concerned doctor. To predict diseases using different ML algorithms like Naïve Bayes, Decision tree, KNN and Random Forest.

5.3 Software Requirements

- Python
- Anaconda
- Jupyter Notebook

- SQLite database

5.4 Hardware Requirements

- Processor: Intel Core i5
- RAM: 8GB
- OS: Windows

5.5 Libraries:

- Tkinter- Tkinter is a library of python used often by everyone. It is a library that is used to create GUI-based applications easily. It contains so many widgets like radio buttons, text fields, and so on. We have used this for creating an account registration screen, log-in or register screen, and prediction interface which is a GUI-based application.
- Sklearn- Scikit Learn also known as sklearn is an open-source library for python programming used for implementing machine learning algorithms. It features various classification, clustering, and regression machine learning algorithms. In this, it is used for importing machine learning models, getting accuracy, get confusion matrix.
- Pandas- Library of python which can be used easily. It gives speedy results and is also easily understandable. It is a library that can be used without any cost. We have used it for data analysis and to read the dataset. It is built on top of another package named Numpy, which provides support for multi- dimensional arrays. The two primary data structures of pandas are Series (1- dimensional) and DataFrame (2-dimensional). we mainly use dataframe in our machine learning task.
- Matplotlib- A library of python used for visualizing the data using graphs, scatterplots, and so on. Here, we have used it for data visualization. It is a cross-platform library for making 2D plots from data in arrays. It provides an object-oriented API that helps in embedding plots in applications using Python GUI

  toolkits such as PyQt, WxPythonotTkinter. It can be used in Python and IPython shells, Jupyter notebook and web application servers also.
- Seaborn - Seaborn is a data visualization library built on top of matplotlib and closely integrated with pandas data structures in Python. Visualization is the central part of Seaborn which helps in exploration and understanding of data.

- Numpy- Library of python used for arrays computation. It has so many functions. We have used this module to change a 2-dimensional array into a contiguous flattened array by using the ravel function.

- Pandas Profiling- This is a library of python which can be used by anyone free of cost. It is used for data analysis. We have used this for getting the report of the dataset.

- SQLite database- SQLite is a C library that provides a lightweight disk-based database that doesn't require a separate server process and allows accessing the database using a nonstandard variant of the SQL query language. Some applications can use SQLite for internal data storage. We can use SQLite in python. SQLite3 can be integrated with Python using sqlite3 module, which was written by Gerhard Haring. It provides an SQL interface compliant with the DB-API 2.0 specification described by PEP 249. You do not need to install this module separately because it is shipped by default along with Python version 2.5. x onwards.

- Pyttsx3 – pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline, and is compatible with both Python 2 and Python 3. An application invokes the pyttsx3.init() factory function to get a reference to a pyttsx3.

- Accuracy Score- Accuracy score is one of the metrics for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition:

  Accuracy=Number of correct predictions/Total number of predictions

- Confusion Matrix - A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model. This gives us a holistic view of how well our classification model is performing and what kinds of errors it is making. For a binary classification problem, we would have a 2 x 2 matrix as shown below in

  Let's decipher the matrix: The target variable has two values: Positive or Negative The column represents the actual values of target variables The row represents the predicted values of target variables

- Precision – Precision tells how many of the correctly predicted cases actually turned out to be positive. Here's how to calculate Precision: Precision = TP/TP+FP

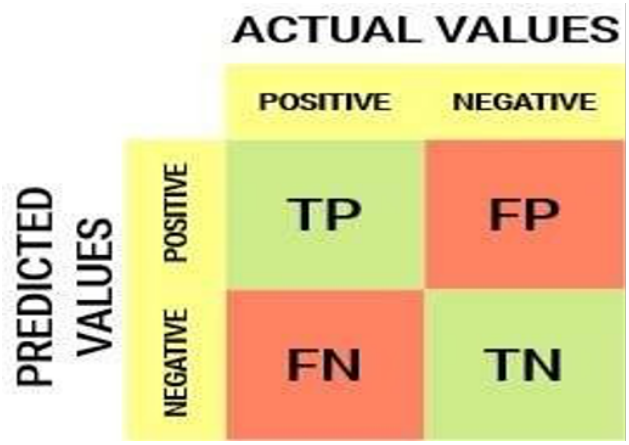

Figure 5.1: Confusion matrix Image

- Recall – Recall tells us how many of the actual positive cases we were able to predicted correctly with our model. And here's how we can calculate Recall: Recall = TP/TP+FN
- F1 SCORE - The F1 Score is the 2*((precision*recall)/ (precision + recall)). It is also called the F Score or the F Measure. Put another way, the F1 score conveys the balance between the precision and the recall.
- Cross validation score – It is a technique used in machine learning to evaluate the performance of a model on unseen data. It involves dividing the available data into multiple folds or subsets, using one of these folds as a validation set, and training the model on the remaining folds. This process is repeated multiple times, each time using a different fold as the validation set. Finally, the results from each validation step are averaged to produce a more robust estimate of the model's performance.

Chapter 6

Implementation

6.1 Development And Deployment Setup

We have developed a GUI application using 4 machine learning algorithms. The main aim of the project is to detect the disease at an early stage so that they can prevent the disease at any early stage rather than curing at later stages. In our model there are nearly 133 symptoms and 41 diseases which are mentioned below.

These are the symptoms and the diseases. First using pandas, we will load our training dataset. Using pandas profiling we will visualize entire our training dataset we get to know each column details precisely how many null values are there, how much memory it consumes, how many duplicate values are there, description of each column, mathematical values, visualization and graphs. Next, we will train our model with 4 machine learning algorithms and create a GUI application and store the details in the database. When a user uses our GUI application, we will store their details in our database like their name and the symptoms they are experiencing and the disease they have. The disease name will be stored in our database based on few parameters like, it will consider the disease which has highest accuracy among four algorithms and the disease which is predicted more than ones by our algorithms. So, by observing our database we can identify the symptoms and diseases they are suffering from and we can report to the user so that they can get a proper treatment according to the disease they are suffering from. So, from this model he can cure his disease at an early stage. The second model is the chatbot we developed it using decision tree algorithm. The chatbot will then inquire about any symptoms similar to those the user has described, as this combination is what creates sickness. The final output of our chatbot will be the disease name, a brief description of the disease, and the precautions. Here the user needs to enter his name and the primary symptom he is experiencing next he can enter from how many days he is

experiencing those symptoms after that our model ask to validate the symptoms that can be present based on the primary symptom. Our model predicts those other symptoms based on the primary symptom. After the user validates our model will predict the disease he has based on the symptoms and it gives a clear explanation about the disease and it says about the severity of the disease like whether he has to consult the doctor immediately or not and the precautions to be taken. This all will be the output for our second model. The speciality of this model is here the user need to enter one symptom rest all our model will take care.Jupyter notebook is needed to develop and deploy the two models.

6.2 Testing

For testing our model, we have a test dataset, after our model get trained from training dataset when user enter the symptoms then it will give the output based on the training it received. We will compare those prediction with the actual predictions. To compare how accurate, it is we have different classification metrices. Based on those scores we can tell how accurate our model; how good it is predicting correct output to the given input. Here we have used different classification metrices like accuracy score, confusion matrix, cross validation score, precision score, recall score, f1 score. This is our testing strategy for GUI application. For chatbot also we will try different times and observe how our model gives output to different inputs and verify with the classification metrices again.

6.3 Types of Tests

Unit testing Unit testing This type of testing involves designing test cases that verify that the program's inner logic is working right and that inputs to the program yield the right outputs. All decision branches and internal code flow need to be tested. It is testing of individual software units of the application .it is done after completion of an individual unit before integration. This is a structural testing, based on its knowledge of the construction and invasive. The unit tests execute simple tests of a component and test a specific business process, application, and or system configuration. Unit tests assure that every different path of a business process executes correctly according to the documented specifications and has well-defined inputs and expected results.

Integration testing Integration tests are written to check integrated software components as to whether they really run as one program. Testing is event-driven and is more interested in general results of screens or fields. Integration tests prove that although the components were satisfied

individually, as demonstrated by successfully unit testing, composite components are correct and consistent. Integration testing is targeted specifically towards exposing problems that arise out of the interaction of components. Software integration testing is incremental integration testing two or more integrated software components on a single platform for purposes of producing failures caused by interface defects. The integration test checks whether 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 of the test cases mentioned above passed successfully. No defects encountered.

Functional test Functional tests provide controlled demonstrations that functions tested are available as stipulated by business and technical requirements, system documentation and user manuals. Functional testing is around 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/Processes: interfaced systems or processes must be called. Organization and preparation of functional tests is based on requirements, key functions, or special test cases. Furthermore, systematic coverage relevant to identify Business process flows; data fields, predefined processes, and successive processes should be considered during testing. Prior to the functional testing completion, additional tests are found and the effective value of the existing tests is determined.

System Test System testing tests all aspects to ensure that the overall integrated software system complies with all of its requirements. This allows a configuration to be tested in a known, predictable way. A configuration-oriented system integration test would be an example of system testing. System testing is based on process descriptions and flows, which are only stressing out the pre-driven process links and integration points.

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 those areas that cannot be reached from the black box level.

Black Box Testing Black Box Testing involves running tests on the software without knowing anything about its inner workings, structure, or language of the module being tested. Black box tests, like most other kinds of testing, must be based on some clear source document, such as specification or requirements document, such as specification or requirements document. It is a black box testing where the software under test is treated like a black box. you are not able to ”see” inside it. The test will present inputs and respond to outputs without regard to the inner workings of the software.

Unit Testing: Unit testing is generally done as part of an integrated code and unit test phase of the software life cycle, though it is common in practice for code and unit testing to take place in two separate phases.

Acceptance Testing User Acceptance Testing: is an important phase of any project and involves considerable participation from the end user. It also ensures that the system satisfies functional requirements.

Test Results: All the above test cases pass all with no defects.

Chapter 7

Conclusion

The project presented the technique of predicting the disease based on the symptoms of an individual patient. Once the disease is predicted, we could easily manage the medicine resources required for the treatment. Doctors and medical professionals are always required in case of an emergency. Our prediction system will be very helpful for finding the disease based on the symptoms in the early stage and get the correct diagnosis of a disease. This also helps in reduction of the cost and give the correct and fast result. The project presented the technique of predicting the disease based on the symptoms of an individual patient. Almost all the ML models gave good accuracy values but the most accurate one is selected and the disease given by it is considered as the disease of an individual. Once the disease is predicted, we could easily manage the medicine resources required for the treatment. This project would help in lowering the cost required in dealing with the disease and would also make the recovery process easy. This study's main objective is to employ a suitable machine learning algorithm to identify the condition from patient-reported symptoms. Using four machine learning algorithms, we were able to predict outcomes in this study with a mean accuracy of more than 98 percent.In terms of accuracy and reliability, this is a considerable advance over past research, making this system more beneficial to users than the current one for this task. It also saves the user's data and the name of the patient's disease in a database that may be used as a historical record and will be utilised for future treatments, making it simpler to monitor the patient's health. We also developed a graphical user interface (GUI) to improve user engagement with the system. This study illustrates the potential of a machine-learning method for disease prediction using a variety of models and parameters. As a result of anyone being able to utilize our system, we can state that it has no user threshold.

Chapter 8

Future Work

There are several potential enhancements that might be investigated in order to diversify the research by learning about and taking into consideration additional qualities. The subsequent tasks must be completed in the future. There are plans to use more voting mechanisms, different discretization techniques, and classification algorithms overall. Would like to employ many rules, such as the association rule, and numerous algorithms, such as logistic regression and clustering algorithms. In the future, willing to apply filter-based feature selection techniques to get more relevant and useful results.A web page which gets symptoms from the user and give the disease as an output can be implemented. And also, Naïve Bayes, accuracy depends on the symptoms given by the user in tkinter. If we normally fit the Naive Bayes model into the dataset, it is showing 100 percent accurate. The reason for this should be found and solve it.

References

1. A. Durga Praveen et al. “Intelligent Liver Disease Prediction system using Machine Learning Models” ,vol 702. Springer, Singapore. 5 Jan, 2021.

2. Akash C. Jamgade, Prof. S. D. Zade,”Disease Prediction using Machine Learning”, International Research Journal of Engineering and Technology Volume: 06 Issue: 05 May 2019.

3. Ahan Chatterjee et al. “A Machine Learning Approach to prevent cancer”, DOI: 10.4018/978-1-7998-2742-9.ch007, 2021.

4. Goutam Chakraboty et al. “Predicting the Risk of Chronic Kidney Disease using Machine Learning Algorithm”, 11(1), 28 December, 2020 202.

5. G. Parthiban, S.K.Srivasta ”Applying Machine learning methods in Diagnosing Heart disease for Diabetic Patients” International Journal of Applied Information Systems (IJAIS) – ISSN: 2249-0868 Foundation of Computer Science FCS, New York, USA Volume 3– No.7, August 2012.

6. Sejin Park et al. “Machine-Learning-Based Elderly Stroke Monitoring System Using Electroencephalography Vital Signals”, 2021, 11(4), 1761.

7. Siddhika Arunachalam,” Cardiovascular Disease Prediction Model using Machine Learning Algorithms”, International Journal for Research in Applied Science, Engineering Technology ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429 Volume 8 Issue VI June 2020.

8. Upendra Kumar, “Applications of Machine Learning In Disease Pre-Screening”, 10.4018/978-1-7998-7705- 9.ch049, 2021.

9. Harshit Anand et al. “Hridaya Kalp: A Prototype for Second Generation Chronic Heart Disease Detection and Classification” ,31 July,2020.

10. Isha Pandya et al, “Prediction of Heart Disease Using Machine Learning Algorithms”, 2018.