**A**

PROJECT REPORT ON

**HEART DISEASE PREDICTIONS USING**

**BIO INSPIRED ALGORITHM**

Submitted in partial fulfillment of the

requirements for the award of the degree of

**BACHELOR OF TECHNOLOGY**

**BY**

**Ms. GIREESHA CHETLAPALLI**

**(Regd.No:21701A3308)**

Under the Guidance of

**Mr. k. Lokeshwar Reddy**

###### Assistant professor



**DEPARTMENT OF**

**ARTIFICIAL INTELLIGENCE & MACHINE LEARNING**

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES

###### (AUTONOMOUS)

NEW BOYANAPALLI-516126, RAJAMPET (A.P)

(Accredited A-Grade by NAAC, Bangalore)

(Approved by A.I.C.T.E, New Delhi &Affiliated to J.N.T.U.A, ANANTHAPURAMU)

**2021-2025\*\*\***

###### ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES (AUTONOMOUS) NEW BOYANAPALLI, RAJAMPET-516126



###### Affiliated to

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR,

ANANTHAPURAMU

**DEPARTMENT OF**

**ARTIFICIAL INTELLIGENCE & MACHINE LEARNING**

**CERTIFICATE**

This is to certify that the project work entitled **“ Heart disease prediction using bio Inspired Algorithms”** is the bonafide work carried out by **Ms. GIREESHA CHETLAPALLI, Regd.No:21701A3308** is submitted in the partial fulfillment of the requirements for the award of degree of Bachelor Of Technology during the year 2021-2025.

**Guide Head of the Department**

**Mrs.S.Surekha Dr.T.Hari Krishna**

**Asst professor Professor and Head**

**Department of AI&ML Department of AI&ML**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |

**TABLE OF CONTENT**

## S.NO TOPIC PAGE NO

### 1. INTRODUCTION 1

1.1 About Industry or Organization Details 2

1.2 My Personal Benefits 4

1.3 Objective of the Project 4 1.4 Limitations of Project 4

### 2. SYSTEM ANALYSIS 5

2.1 Introduction 6

2.2 Existing System 6

2.3 Disadvantages of Existing System 9

2.4 Proposed System 9

2.5 Advantages over Existing System 12

### 3. SYSTEM SPECIFICATION 13

3.1 Hardware Requirement Specification 14

3.2 Software Requirement Specification 14

### 4. SYSTEM DESIGN 16

4.1 System Architecture 17

4.2 Module Flow Diagrams 18

v

### 5. IMPLEMENTATION AND RESULTS 19

5.1 Introduction 20

5.2 Implementation of Key Functions 20

5.3 Method of Implementation(CODING) 21 5.4 Output Screens and Result Analysis 32

### 6. TESTING AND VALIDATION 37

6.1 Introduction 38

6.2 Design of Test cases and Scenarios 39

6.3 Validation 39

1. **CONCLUSION 40**

7.1 Conclusion 41

## REFERENCES 42

vi

**LIST OF FIGURES**

**Page**

**S .NO. Figure Name of the figure Number**

1. 3.1.1 SVM Algorithm Flowchart 10
2. 3.2.1 Bio-Inspired Algorithm Flowchart 14
3. 3.2.2 System Architecture Flow 15
4. 4.1.1 preprocess of the algorithm 17
5. 4.2.1 Architecture 18
6. 6.1.1 Home Screen 32
7. 6.1.2 Dataset File Upload 32
8. 6.1.3 Dataset Uploaded Page 33
9. 6.1.4 Dun Genetic Algorithm 33
10. 6.1.5 Run BAT Algorithm 34
11. 6.1.6 Run BEE Algorithm 34
12. 6.1.7 Upload Test File 35
13. 6.1.8 Predicted Diseases Stages 35
14. 6.1.9 Accuracy Graph 36

vii

## LIST OF ABBREVIATIONS

|  |  |
| --- | --- |
| **EWR** | Electronic Wellbeing Records |
| **CNN** | Convolutional Neural Network |
| **CP** | Chest Pain |
| **RESTBPS** | Resting Blood Pressure |
| **CHOL** | Cholestrol |
| **FBS** | Fasting Blood Pressure |
| **RESTECG** | Resting Electrocardiography |
| **EXANG** | Exercise Induced Angina |
| **KNN** | K Nearest Neighbours |
| **DSS** | Decision Support System |
| **ANN** | Artificial Neural Network |
| **WEKA** | Waikato Environment for Knowledge Analysis |
| **CAD** | Coronary Artery Disease |
| **SVM** | System Virtual Machine |
| **WHO** | World Health Organization |
| **ACO** | Ant Colony Optimization |
| **ABC** | Artificial Bee Colony |

viii

### ABSTRACT

Heart related diseases or Cardiovascular Diseases (CVDs) are the main reason for a huge number of deaths in the world over the last many decades and has emerged as the most life- threatening disease, not only in India but also in the whole world. Prediction of cardiovascular disease is a critical challenge in the area of clinical data analysis. So, there's a need of dependable, accurate and possible system to diagnose similar diseases in time for proper treatment. Machine Learning algorithms and approaches have been applied to various medical datasets to automate the analysis of large and complex data. multiple experimenters, in recent times, have been using several machine learning approaches to help the health care industry and the professionals in the diagnosis of heart related diseases. This project presents a review of various models based on like algorithms and approaches and analyzes their performance. The main aim of this design is to give an effective algorithm to predict heart disease. So, at the end we compare our algorithm (Genetic algorithm) with BAT and BEE algorithms and we prove that the produced algorithm is effective one among all. Also, we forecast the output by taking some random data.

**CHAPTER – 1**

**INTRODUCTION**

### 1.1 ABOUT INDUSTRY OR ORGANIZATION DETAILS

As indicated in a report by McKinsey, half of Americans have at least one persistent sickness, and 80% of American clinical consideration expense is spent on persistent illness treatment. With the improvement of expectations for everyday comforts, the rate of persistent sickness is expanding. The US has spent a normal of 2.7 trillion used every year on persistent infection treatment. This sum includes 18% of the whole yearly gross domestic product of the US. The medical services issue of ongoing illnesses is likewise vital in numerous different nations. In China, persistent sicknesses are the primary driver of death, as indicated by a Chinese report on sustenance and persistent illnesses in 2015, 86.6% of passing are brought about by constant infections. Thus, performing risk evaluations for constant diseases is fundamental. With the development in clinical information, gathering electronic wellbeing records (EWR) is progressively advantageous. Furthermore, first introduced a bio-enlivened elite execution heterogeneous vehicular telematics worldview, with the end goal that the assortment of versatile clients' wellbeing related continuous large information can be accomplished with the organization of cutting edge heterogeneous vehicular organizations. Patients' factual data, test results furthermore, illness history are kept in the EWR, empowering us to distinguish potential information driven answers for decrease the expenses of clinical contextual investigations. Proposed an ideal large information sharing calculation to deal with to entangle informational collection in telehealth with cloud strategies. One of the applications is to identify high-risk patients which can be used to diminish clinical expense since high-risk patients frequently require costly medical services. Also, in the principal paper proposing medical services digital actual framework, it creatively presented the idea of expectation based medical care applications, counting wellbeing risk evaluation. Expectation utilizing customary sickness risk models ordinarily implies an ai calculation (e.g., strategic relapse and relapse investigation, and so forth), and particularly a directed learning calculation by the utilization of preparing information with names to prepare the model. In the test set, patients can be ordered into gatherings of either high-chance or generally safe. These models are important in clinical circumstances and are generally considered. Not with standing, these plans have the accompanying qualities and deformities.The informational collection is ordinarily little, for patients and infections with explicit circumstances, the attributes are chosen through experience. In any case, these prechosen qualities perhaps not fulfill the progressions in the illness and its affecting elements. With the improvement of enormous information examination innovation, more consideration has been

paid to illness forecast from the viewpoint of huge information investigation, different explores have been led by choosing the qualities naturally from countless information to work on the exactness of hazard order instead of the recently chosen qualities. Nonetheless, that current work for the most part thought of organized information. For unstructured information, for instance, utilizing convolutional brain organization (CNN) to separate text attributes consequently has previously drawn in wide consideration and additionally accomplished awesome outcomes. Be that as it may, to the best of our insight, none of past work handle Chinese clinical text information by CNN. Thigh-risk patients which can be used to diminish clinical expense since high-risk patients frequently require costly medical services. Also, in the principal paper proposing medical services digital actual framework, it creatively presented the idea of expectation based medical care applications, counting wellbeing risk evaluation. Expectation utilizing customary sickness risk models ordinarily implies an AI calculation (e.g., strategic relapse and relapse investigation, and so forth), and particularly a directed learning calculation by the utilization of preparing information with names to prepare the model.

In the test set, patients can be ordered into gatherings of either high-chance or generallysafe. These models are important in clinical circumstances and are generally considered. Notwithstanding, these plans have the accompanying qualities and deformities. The informational collection is ordinarily little, for patients and infections with explicit circumstances, the attributes are chosen through experience. In any case, these pre-chosen qualities perhaps not fulfill the progressions in the illness and its affecting elements. With the improvement of enormous information examination innovation, more consideration has been paid to illness forecast from the viewpoint of huge information investigation, different explores have been led by choosing the qualities naturally from countless information to work on the exactness of hazard order, instead of the recently chosen qualities. Nonetheless, that current work for the most part thought of organized information. For unstructured information, for instance, utilizing convolutional brain organization (CNN) to separate text attributes consequently has previously drawn in wide consideration and additionally accomplished awesome outcomes. Be that as it may, to the best of our insight, none of past work handles Chinese clinical text information by CNN.

### 1.2 MY PERSONAL BENEFITS

The main provocation of doing this disquisition is to present a heart complaint prophecy model for the prophecy of circumstance of heart complaint. Further, this disquisition work is aimed towards relating the stylish bracket algorithm for relating the possibility of heart complaint in a case. This work is justified by performing a relative study and analysis using three bracket algorithms videlicet heritable, club and freak are used at different situations of evaluations Although these are generally used Bio Inspired algorithms, the heart complaint prophecy is a vital task involving topmost possible delicacy. Hence, the three algorithms are estimated at multitudinous situations and types of evaluation strategies. This will give inquiries and medical practitioners to establish a better.

### 1.3 OBJECTIVE OF THE PROJECT

The ideal of our design heart disease prediction using bio inspired algorithms is to determine if a case should be diagnosed with heart disease or not, which is a double outgrowth.

So, Positive result = 1, the case will be diagnosed with heart complaint and Negative result = 0, the case won't be diagnosed with heart complaint.

We've to find this with classification model which has the topmost accuracy and identify correlations in our data. Eventually, we also have to determine which features are the most influential in our heart complaint diagnosis.

### 1.4 LIMITATIONS OF THE PROJECT

The major challenge in heart complaint is its discovery. There are instruments available which can predict heart complaint but either it's precious or aren't effective to calculate chance of heart complaint in mortal. Beforehand discovery of cardiac conditions can drop the mortality rate and overall complications. still, it isn't possible to cover cases every day in all cases directly and discussion of a case for 24 hours by a doctor isn't available since it requires further insight, time and expertise. Since we've a good quantum of data in moment’s world, we can use various machine learning algorithms to dissect the data for hidden patterns. The hidden patterns can be used forhealth diagnosis in medicinal data.

**CHAPTER – 2**

**SYSTEM ANALYSIS**

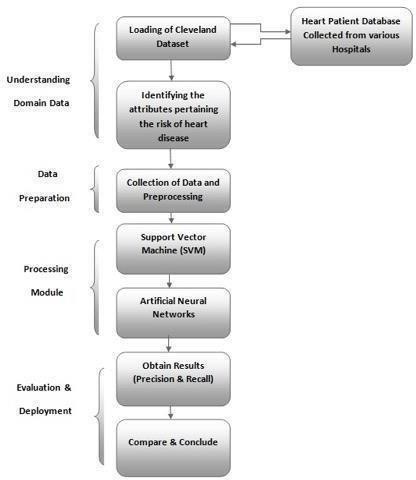
### 2.1 INTRODUCTION

Heart related diseases or Cardiovascular Diseases(CVDs) are the main reason for a huge number of deaths in the world over the last many decades and has emerged as the most life- threatening disease, not only in India but also in the whole world. Prediction of cardiovascular disease is a critical challenge in the area of clinical data analysis. So, there's a need of dependable, accurate and possible system to diagnose similar diseases in time for proper treatment. This project presents a review of various models based on like algorithms and approaches and analyzes their performance. The main aim of this design is to give an effective algorithm to predict heart disease. So, at the end we compare our algorithm (Genetic algorithm) with BAT and BEE algorithms and we prove that the produced algorithm is effective one among all. Also, we forecast the output by taking some random data. Causes of heart infection are bacteria, viruses, parasites. Types of heart disease: Cardiac arrest, Hypertension, Coronary artery disease, Heart failure, Heart infection, Congenital heart disease, Slow heartbeat, Stroke type heart disease, angina pectoris. Now a days there are toomany automated techniques to detect the heart disease like data mining, machine learning, deep learning, and using bio inspired alogorithms.

**2.1 EXISTING SYSTEM:**

**SVM ALGORITHM**

In machine literacy, support- vector machines (SVMs, also support- vector networks) are supervised literacy models with associated literacy algorithms that dissect data for bracket and retrogression analysis. Developed at AT&T Bell Laboratories by Vladimir Vapnik with associates (Boser etal., 1992, Guyon etal., 1993, Cortes and Vapnik, 1995, Vapnik etal., 1997 (citation demanded)) SVMs are one of the most robust vaticination styles, being grounded on statistical literacy fabrics or VC proposition proposed by Vapnik (1982, 1995) and Chervonenkis (1974). Given a set of training exemplifications, each marked as belonging to one of two orders, an SVM training algorithm builds a model that assigns new exemplifications to one order or the other, making it a non-probabilistic double direct classifier (although styles similar as Platt spanning live to use SVM in a probabilistic bracket setting). SVM maps training exemplifications to points in space so as to maximize the range of the gap between the two orders. New exemplifications are also counterplotted into that same space and prognosticated to belong to a order grounded on which side of the gap they fall. In addition to performing direct bracket, SVMs can efficiently perform anon-linear bracket using what's called the kernel trick, implicitly mapping their inputs into high- dimensional point spaces. When data are un labelled, supervised literacy isn't possible, and an unsupervised literacy approach is needed, which attempts to find natural clustering of the data to groups, and also collude new data to these formed groups. The support- vector clustering algorithm, created byHava Siegelmann and Vladimir Vapnik, applies the statistics of support vectors, developed in the support vector machines algorithm, to classify unlabeled data. Classifying data is a common task in machine literacy. Suppose some given data points each belong to one of two classes, and the thing is to decide which class a new data point will be in. In the case of support- vector machines, a data point is viewed as a {display style p} p- dimensional vector (a list of {display style p} p figures), and we want to know whether we canseparate similar points with a {display style (p- 1)} (p- 1)- dimensional hyperplane. This is called a direct classifier. There are numerous hyperplanes that might classify the data. One reasonable choice as the stylish hyperplane is the bone that represents the largest separation, or periphery, between the two classes. So, we choose the hyperplane so that the distance from it to the nearest data point on each side is maximized. However, it's known as the outside- periphery hyperplane and the direct classifier it defines is known as an outside-periphery classifier; or equally, the perceptron of optimal stability (citation demanded), If such a hyperplane exists.



#### Figure 2.1.1 SVM Algorithm Flowchart

Further formally, a support- vector machine constructs a hyperplane or set of hyperplanes in a high- or horizonless- dimensional space, which can be used for bracket, retrogression, or other tasks like outlier’s discovery. Intimately, a good separation is achieved by the hyperplane that has the largest distance to the nearest training- data point of any class (so-called functional periphery), since in general the larger the periphery, the lower the conception error of the classifier.Whereas the original problem may be stated in a finite- dimensional space, it frequentlyhappens that the sets to distinguish aren't linearly divisible in that space. For this reason, it was proposed that the original finite- dimensional space be counterplotted into a much advanced- dimensional space, presumably making the separation easier in that space.

To keep the computational cargo reasonable, the mappings used by SVM schemes are designed to insure that fleck products of dyads of input data vectors may be reckoned fluently in terms of the variables in the original space, by defining them in terms of a kernel function {display style k(x, y)}{ display style k( x, y)} named to suit the problem. The hyperplanes in the advanced- dimensional space are defined as the set of points whose fleck product with a vector in that space is constant, where such a set of vectors is an orthogonal (and therefore minimum) set of vectors that defines a hyperplane. The vectors defining the hyperplanes can be chosen to be direct combinations with parameters {display style nascence, {i}} nascence, {i} of images of point vectors {display style x, {i}} x, {i} that do in the data base. With this choice of a hyperplane, the points {display style x} x in the point space that are counterplotted into the hyperplane are defined by the relation {display style textbook style sum, {i} nascence, {i} k (x, {i}, x) = {textbook {constant}}.} {display style textbook style sum, {i} nascence, {i} k (x, {i}, x) = {textbook {constant}}.} Note that if {display style k (x, y)} {display style k (x, y)} becomes small as {display style y} y grows further down from {display style x} x, each term in the sum measures the degree of closeness of the test point {display style x} x to the corresponding data base point {display style x, {i}} x, {i}. In this way, the sum of kernels over can be used to measure the relative nearness of each test point to the data points forming in one or the other of the sets to be discerned. Note the fact that the set of points {display style x} x counterplotted into any hyperplane can be relatively sophisticated as a result, allowing much more complex demarcation between sets that aren't convex at each in the original space. Heart conditions have surfaced as one of the most prominent Cause of death each aroundthe world. According to World Health Organization, heart related conditions are responsible for the taking17.7 million lives every time, 31 of all global deaths. In India too, heart related conditions have come the leading cause of mortality. Heart conditions have killed1.

### 2.2 Disadvantages of Existing System

2.3.1.1 These tests are too expensive.

2.3.1.2 Prediction of cardiovascular disease results is not accurate.

2.3.1.3 Cannot handle enormous datasets for patient records.

2.3.1.4 Data mining techniques does not help to provide effective decision making.

**2.3 PROPOSED SYSTEM:**

After evaluating the results from the existing methodologies, we've used python operations to perform heart disease classification for the data attained from the UCI repository. It provides an easy- to- use visual representation of the dataset, working environment and building the predictive analytics. Genetic Algorithm process starts from a preprocessing data phase followed by feature selection based on data cleaning, classification of modelling performance evaluation. also, BAT and BEE algorithms are used to improve the accuracy of the result.

Dimensionality reduction involves opting a mathematical representation such that one can relate the majority of, but not all, the variance within the given data, thereby including only most significant information. The data considered for a task or a problem, may consists of a lot of attributes or dimensions, but not all of these attributes may equally impact the output. A large number of attributes, or features, may affect the computational complexity and may indeed lead to overfitting which leads to poor results. therefore, dimensionality reduction is a very important step considered while building any model. Dimensionality reduction is generally achieved by two methods- feature extraction and feature selection.

To implement this algorithms we are using Heart disease dataset which contains 14 attributes and 4 class labels where 0 refers to No heart Disease and 1 refers to stage1 disease and 2 and 3 refers stage 3 and 4 disease.

Below are some values from dataset to train algorithms age,sex,cp,trestbps,chol,fbs,restecg,thalach,exang,oldpeak,slope,ca,thal,class

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

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

67.0,1.0,4.0,120.0,229.0,0.0,2.0,129.0,1.0,2.6,2.0,2.0,7.0,1

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

First records contain dataset column names and remaining records are the values of dataset. In last column we have class values as 0, 2, 1 and 3 as disease stage.

Test dataset also contains record values but it will not have class labels and application will apply that test values on train dataset to predict it class labels. Some values from test dataset.

age,sex,cp,trestbps,chol,fbs,restecg,thalach,exang,oldpeak,slope,ca,thal

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

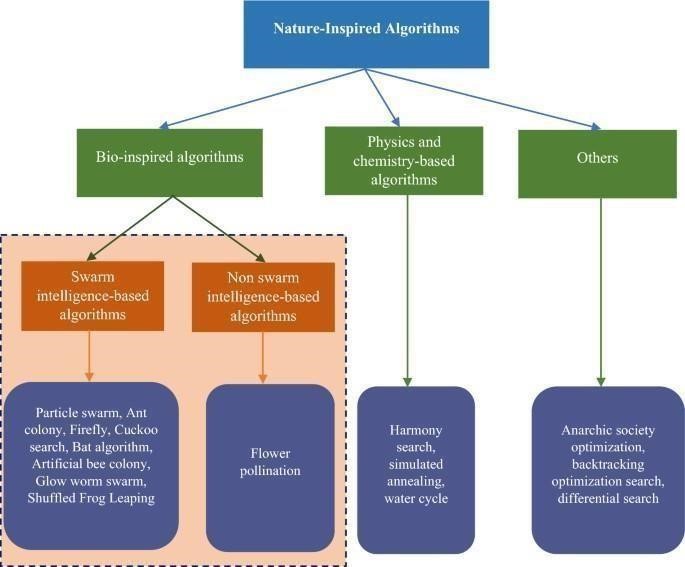
67.0,1.0,4.0,160.0,286.0,0.0,2.0,108.0,1.0,1.5,2.0,3.0,3.0

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

In above test dataset we can see there is no class name and application will predict it.

Heart Disease Prediction Using Bio Inspired Algorithms

In this project student want to detect heart disease from dataset using Bio Inspired 4 features optimizing algorithms such as Genetic Algorithm, Bat, Bee and ACO. Here ACO algorithm is design in python to solve Travelling Salesman Problem to find shortest path and it cannot be implemented with heart disease dataset, so we are implementing 3 algorithms called Genetic, Bat and Bee. Bio inspired algorithms design to optimized features used in dataset for training classification algorithms to increase prediction accuracy, sometime some datasets may have irrelevant values inside dataset and those irrelevant attributes or values may degrade classification accuracy so using optimize algorithms we can reduce features (attribute values) from dataset. This optimize algorithms will be applied on dataset to check whether all values are related to dataset or not, if any attribute found unrelated then it will removed from dataset.



#### Figure: 2.2.1 Bio-Inspired Algorithm Flowchart

The genetic algorithm is a method for solving both constrained and unconstrained optimization problems that is based on natural selection, the process that drives biological evolution. The genetic algorithm repeatedly modifies a population of individual solutions. At each step, the genetic algorithm selects individuals from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, the population "evolves" toward an optimal solution. You can apply the genetic algorithm to solve a variety of optimization problems that are not well suited for standard optimization algorithms, including problems in which the objective function is discontinuous, nondifferentiable, stochastic, or highly nonlinear. The genetic algorithm can address problems of mixed integer programming, where some components are restricted to be integer-valued. Bio-Inspired Genetic Algorithms with Formalized Crossover Operators for Robotic Applications. Genetic algorithms are widely adopted to solve optimization problems in robotic applications. In such safety-critical systems, it is vitally important to formally prove the correctness when genetic algorithms are applied.

In ABC, a colony of artificial probe notions(agents) quest for rich artificial food sources(good results for a given problem). To apply ABC, the considered optimization problem is first converted to the problem of chancing the stylish parameter vector which minimizes an objective function. also, the artificial notions erratically discover a population of original result vectors and also iteratively ameliorate them by employing the strategies moving towards better results by means of a neighbour quest medium while abandoning poor results.

### 2.5 Advantages over Existing System

1. Increased accuracy for effective heart disease diagnosis.
2. Handles roughest(enormous) amount of data using Genetic algorithm and feature selection.
3. Reduce the time complexity of doctors.
4. Cost effective for patients.

## CHAPTER 3

**SYSTEM SPECTIFICATION**

**3.1 SOFTWARE REQUIREMENT SPECIFICATION:**

1. **Programming Language:**

- Choose a programming language suitable for machine learning and Python.

1. **Bio Inspired Algorithms**

- Here we use BAT and BEE algorithms. These algorithms are used to solvethe computer science problems using models and biology

**Operating System** - Windows 10, Windows 11

**Technology** - Python3.7

**3.2 HARDWARE REQUIREMENT SPECIFICATION:**

* + 1. **Processor (CPU) -**

The choice of CPU depends on the scale of the project. For small to mediumsized chatbots, a multi-core CPU can be sufficient. For larger projects with high computational demands, especially if dealing with deep learning models, more powerful CPUs or even server-grade processors may be required.

* + 1. **Graphics Processing Unit (GPU)-**

GPUs excel in parallel processing tasks and are crucial for training and running complex machine learning models, including those used in natural language processing. NVIDIA GPUs are commonly used for deep learning tasks, but the specific GPU requirements depend on the AI framework being utilized.

* + 1. **Random Access Memory (RAM)-**

Sufficient RAM is essential for efficiently running AI models and handling concurrent user interactions. The amount of RAM needed depends on the size of the model and the expected user load. At least 16GB of RAM is recommended for most AI applications, but more may be needed for larger models or heavier workloads.

* + 1. **Storage-**

Adequate storage is necessary for storing the AI models, datasets, and other related files. Solid State Drives (SSDs) are preferred over Hard Disk Drives (HDDs) for faster data access and retrieval.

**Processer** - I5/ Intel Processor

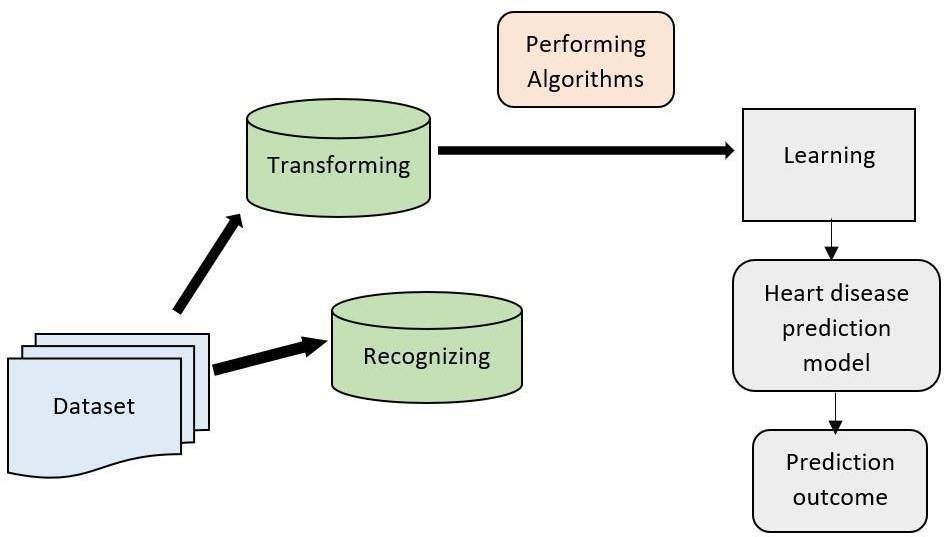
**Ram** - 8 GB

**Hard Disk** - 512 GB

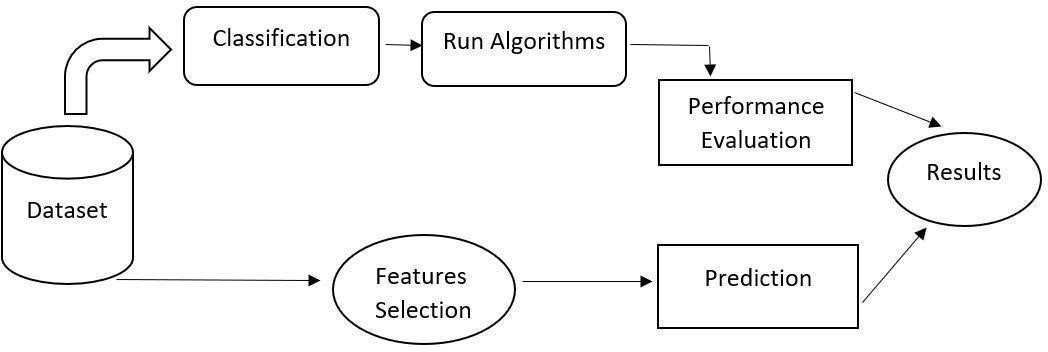
**CHAPTER – 4**

**SYSTEM DESIGN**

### 4.1 SYSTEM ARCHITECTURE



**Fig:4.1.1 preprocess of the algorithm**



**Fig:4.2.1 Architecture**

## CHAPTER 5

**IMPLEMENTATION AND RESULTS**

### 

**5.1 INTRODUCTION**

Health diseases are increasing day by day due to life style and hereditary. In this

aspect, heart disease is the most important cause of demise in the human kind over past few years. The objective of this paper is to predict the Heart Disease by applying Conventional Neural Network using Bio Inspired algorithm. Bio Inspired Algorithms is short for biologically inspired computing, is a field of study which seeks to solve computer science problems using models of biology. Here we are going to Implement these bio inspired optimization algorithms which represent a promising approach for solving complex optimization problems. Here BEE algorithm is an optimization algorithm based on the intelligent foraging behavior of homey bee swarm. And Bat algorithm is a Heuristic algorithm that operates by imitating the echolocation behavior of bats to perform global operations. Both these algorithms are implemented here.

### 5.2 IMPLEMENTATION AND KEY FUNCTIONS

1. **Requirements Gathering:**

In this phase we gather all the requirements from the client, i.e. what are the client expected input, output.

1. **Analysis:**

In this phase based upon the client requirements we prepare one documentation is called

“High Level Design Document”. It contains Abstract, Functional Requirements, Non Functional Requirements, Existing System, Proposed System, SRS.

1. **Design:**

It is difficult to understand the High Level Design Document for all the members, so to understand easily we use “Low Level Design Document”. To design this document we use UML (Unified Modeling Language).

1. **Coding:**

In this phase we develop the coding module by module. After developing all the modules we integrate them.

1. **Testing:**

After developing we have to check weather client requirements are satisfied or not. If not we are again going to develop.

1. **Implementation:**

In testing phase if client requirements are satisfied, we go for implementation. i.e. we need to deploy the application in some server.

1. **Maintenance:**

After deployment, if at all any problems come from the client side; we are providing maintenance for that application.

### 5.3 METHOD OF IMPLEMENTATION (Coding)

**Main.py**

from future import print\_function

from tkinter import messagebox from tkinter import \*

from tkinter import simpledialog

import tkinter

from tkinter import filedialog import matplotlib.pyplot as plt from tkinter.filedialog import

askopenfilename

from sklearn.model\_selection import train\_test\_split

from sklearn.ensemble import

RandomForestClassifier

import os

import re

from sklearn.metrics import accuracy\_score import numpy as np

from sklearn import datasets, linear\_model import pandas as pd from genetic\_selection import

GeneticSelectionCV from sklearn.metrics import classification\_report

from sklearn.metrics import confusion\_matrix

import SwarmPackagePy

from sklearn import svm from sklearn.ensemble import

RandomForestClassifier from BAT import BAT from SwarmPackagePy import testFunctions

as tf

from BEE import BEE main = tkinter.Tk()

main.title("Heart Disease Prediction Using

Bio Inspired Algorithms") main.geometry("1300x1200")

global filename global train global ga\_acc, bat\_acc, bee\_acc

global classifier

def upload(): global filename filename =

filedialog.askopenfilename(initialdir=

"heart\_dataset") pathlabel.config(text=filename)

text.delete('1.0', END)

text.insert(END,filename+" loaded\n");

def prediction(X\_test, cls): #prediction done here

y\_pred = cls.predict(X\_test) for i in range(len(X\_test)):

print("X=%s, Predicted=%s" %

(X\_test[i], y\_pred[i])) return y\_pred

# Function to calculate accuracy

def cal\_accuracy(y\_test, y\_pred, details): cm = confusion\_matrix(y\_test, y\_pred) accuracy = accuracy\_score(y\_test,y\_pred)\*100

text.insert(END,details+"\n\n") text.insert(END,"Accuracy :

"+str(accuracy)+"\n\n")

text.insert(END,"Report :

"+str(classification\_report(y\_test, y\_pred))+"\n")

text.insert(END,"Confusion Matrix :

"+str(cm)+"\n\n\n\n\n") return accuracy

def geneticAlgorithm(): global classifier text.delete('1.0', END) global ga\_acc train = pd.read\_csv(filename)

test = pd.read\_csv('heart\_dataset/test.txt')

test\_X = test.values[:, 0:12] X = train.values[:, 0:12] y = train.values[:, 13] estimator = linear\_model.LogisticRegression(solv er="liblinear", multi\_class="ovr")

selector = GeneticSelectionCV(estimator,

cv=5, verbose=1, scoring="accuracy", max\_features=10, n\_population=50, crossover\_proba=0.5, mutation\_proba=0.2, n\_generations=200,

crossover\_independent\_proba=0.5, mutation\_independent\_proba=0.05, tournament\_size=3, n\_gen\_no\_change=10, caching=True, n\_jobs=-1)

selector = selector.fit(X, y) y\_pred = selector.predict(test\_X) prediction\_data = prediction(test\_X, selector)

ga\_acc = cal\_accuracy(prediction\_data, prediction\_data,'GA Algorithm Accuracy, Classification Report &

Confusion Matrix') classifier = selector

def runBat(): text.delete('1.0', END) global bat\_acc train = pd.read\_csv(filename) alh = BAT(train.values, tf.easom\_function,

-10, 10, 2, 20)

data = alh.get\_agents()

1. = []
2. = []

for i in range(len(data)):

for j in range(len(data[i])):

X.append(data[i][j][0:13])

Y.append(data[i][j][13])

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, Y, test\_size = 0.1, random\_state = 0) cls =

RandomForestClassifier(n\_estimators

=50,max\_depth=2,random\_state=0,cla ss\_weight='balanced')

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

prediction\_data = prediction(X\_test, cls) bat\_acc = cal\_accuracy(y\_test, prediction\_data,'BAT Algorithm Accuracy, Classification Report &

Confusion Matrix')

def runBee(): text.delete('1.0', END) global bee\_acc train = pd.read\_csv(filename)

alh = BEE(train.values, tf.easom\_function,

-10, 10, 2, 20)

data = alh.get\_agents()

1. = []
2. = []

for i in range(len(data)):

for j in range(len(data[i])):

X.append(data[i][j][0:13])

Y.append(data[i][j][13])X\_train,

X\_test, y\_train, y\_test =

train\_test\_split(X, Y, test\_size = 0.1,

random\_state = 0)

cls =

RandomForestClassifier(n\_estimators

=30,max\_depth=2,random\_state=0,cla ss\_weight='balanced')

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

prediction\_data = prediction(X\_test, cls) bee\_acc = cal\_accuracy(y\_test, prediction\_data,'ABE Algorithm

Accuracy, Classification Report &

Confusion Matrix') def predict(): text.delete('1.0', END) filename = filedialog.askopenfilename(initialdir=

"dataset") test = pd.read\_csv(filename) test = test.values[:, 0:12] total = len(test) text.insert(END,filename+" test file loaded\n");

y\_pred = classifier.predict(test) for i in range(len(test)):

print(str(y\_pred[i]))

if str(y\_pred[i]) == '0.0': text.insert(END,"X=%s, Predicted = %s" % (test[i], 'No disease

detected')+"\n\n") if str(y\_pred[i]) == '1.0': text.insert(END,"X=%s, Predicted = %s" % (test[i], 'Stage 1 Disease

Detected')+"\n\n") if str(y\_pred[i]) == '2.0': text.insert(END,"X=%s, Predicted =

%s" % (test[i], 'Stage 2 Disease

Detected')+"\n\n")

if str(y\_pred[i]) == '3.0': text.insert(END,"X=%s, Predicted = %s" % (test[i], 'Stage 3 Disease

Detected')+"\n\n")

if str(y\_pred[i]) == '4.0': text.insert(END,"X=%s, Predicted = %s" % (test[i], 'Stage 4 Disease

Detected')+"\n\n") def graph():

height = [ga\_acc,bat\_acc,bee\_acc] bars = ('Genetic Algorithm','KNN

Algorithm','Decision Tree Algorithm')

y\_pos = np.arange(len(bars))

plt.bar(y\_pos, height) plt.xticks(y\_pos, bars)

plt.show()

def exit(): main.destroy() font = ('times', 16, 'bold') title = Label(main, text='Heart Disease Prediction Using Bio Inspired

Algorithms') title.config(bg='brown', fg='white') title.config(font=font) title.config(height=3, width=120)

title.place(x=0,y=5)

font1 = ('times', 14, 'bold')

uploadButton = Button(main, text="Upload

Heart Disease", command=upload) uploadButton.place(x=50,y=100)

uploadButton.config(font=font1)

pathlabel = Label(main) pathlabel.config(bg='brown', fg='white')

pathlabel.config(font=font1)

pathlabel.place(x=460,y=100)

geneticButton = Button(main, text="Run

Genetic Algorithm",

command=geneticAlgorithm)

geneticButton.place(x=50,y=150) geneticButton.config(font=font1) batButton = Button(main, text="Run BAT

Algorithm", command=runBat) batButton.place(x=330,y=150)

batButton.config(font=font1)

beeButton = Button(main, text="Run BEE

Algorithm", command=runBee) beeButton.place(x=620,y=150)

beeButton.config(font=font1)

predictButton = Button(main, text="Upload

& Predict Test Data",

command=predict)

predictButton.place(x=850,y=150)

predictButton.config(font=font1)

graphButton = Button(main, text="Accuracy

Graph", command=graph) graphButton.place(x=50,y=200)

graphButton.config(font=font1)

exitButton = Button(main, text="Exit", command=exit) exitButton.place(x=330,y=200)

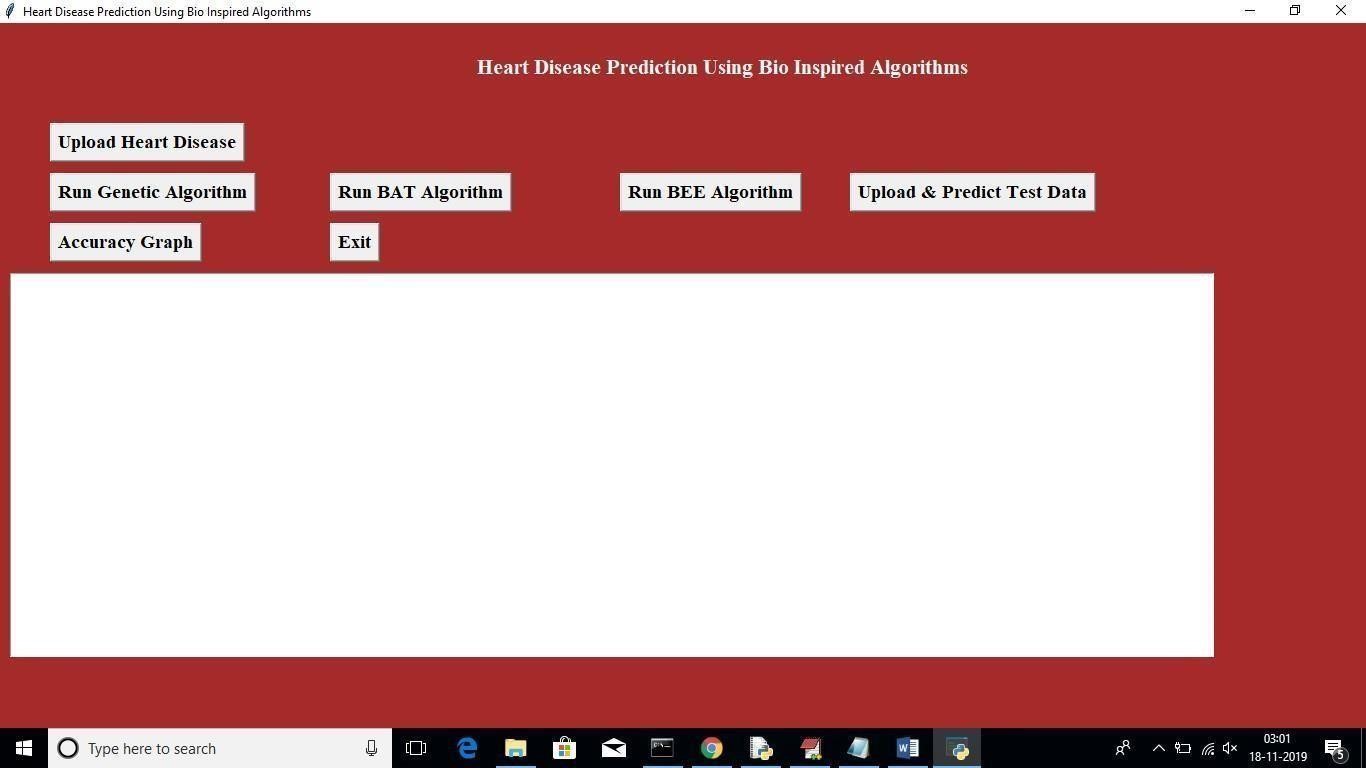
exitButton.config(font=font1)

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

text=Text(main,height=20,width=150) scroll=Scrollbar(text) text.configure(yscrollcommand=scroll.set) text.place(x=10,y=250) text.config(font=font1) main.config(bg='brown') main.mainloop()

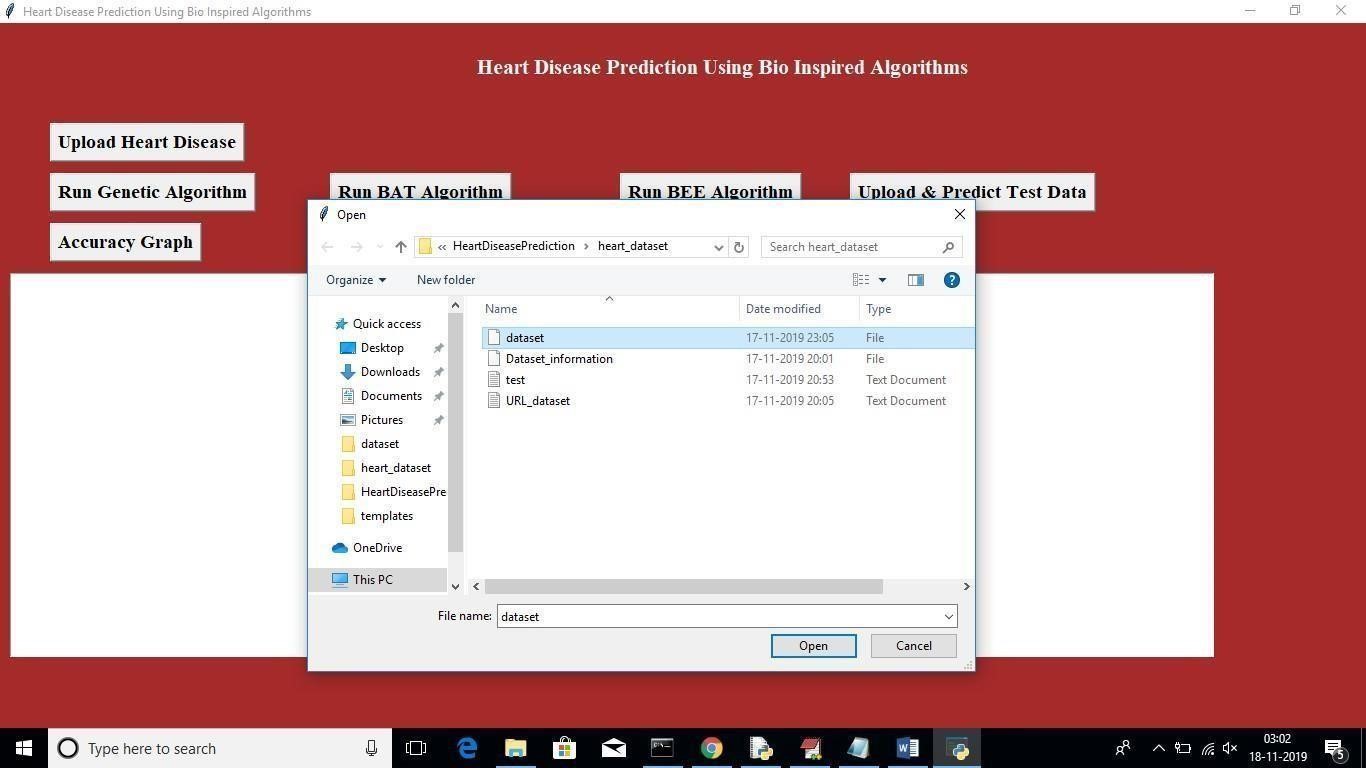
**5.4 OUTPUT SCREENSHOTS AND RESULT ANALYSIS:**

To run this project double click on ‘run.bat’ file to get below screen



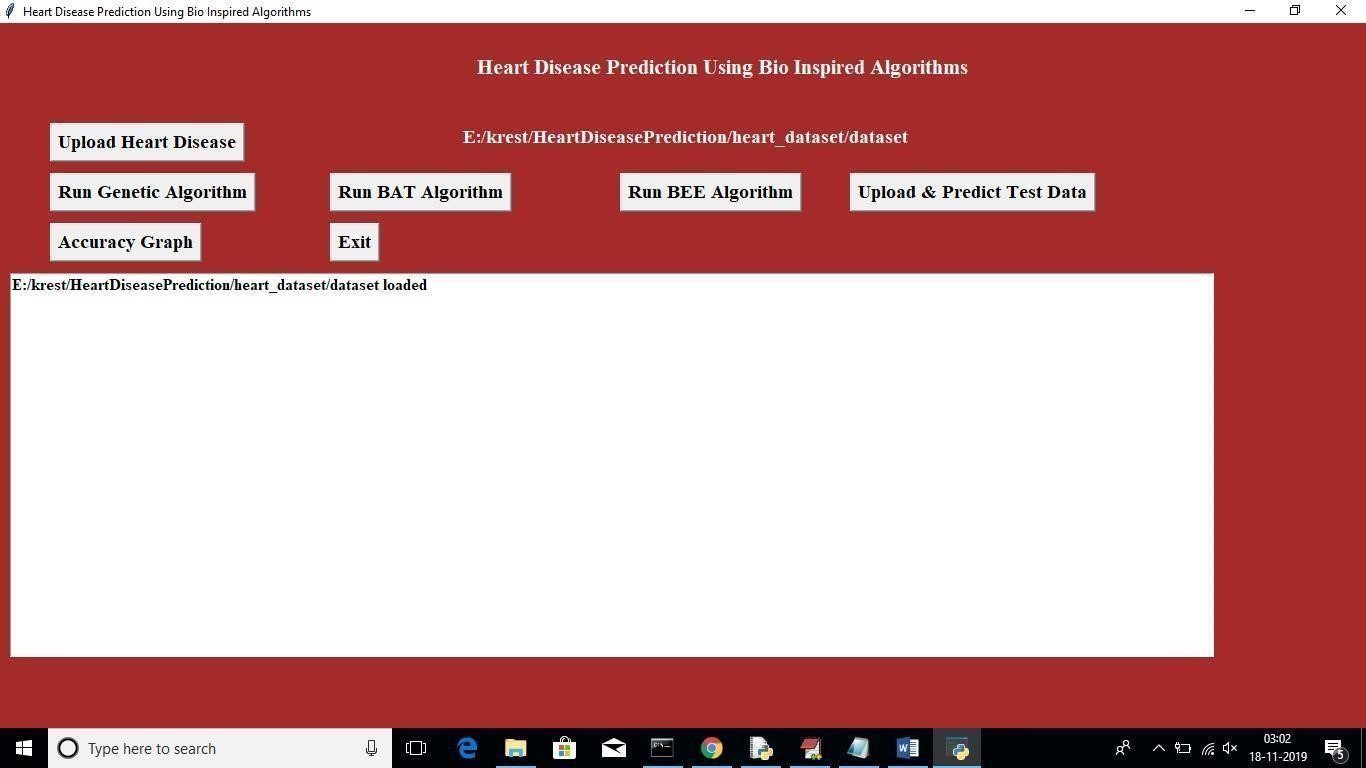
#### Fig 6.1.1: Home Screen

In above screen click on ‘Upload Heart Disease’ button and upload heart disease dataset. See below screen



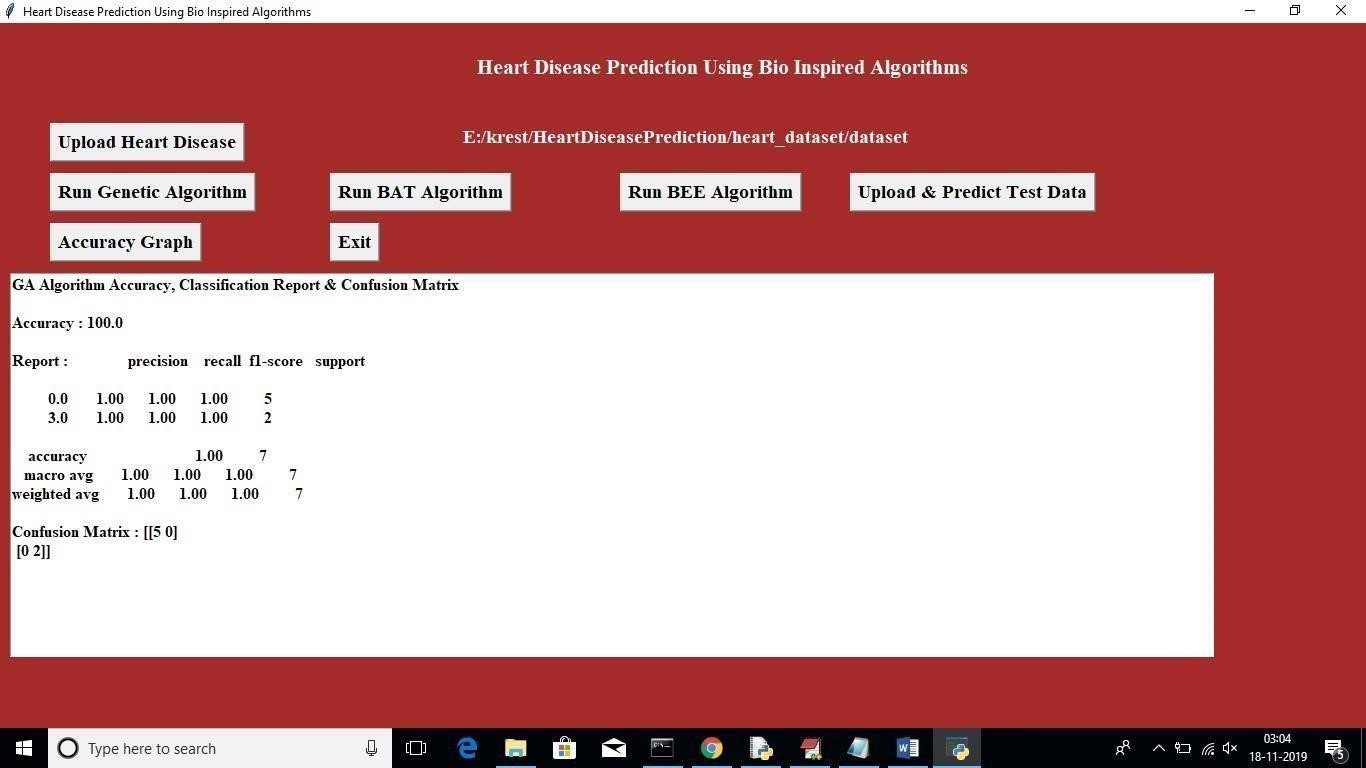
#### Fig 6.1.2: Dataset File Upload

In above screen uploading dataset file, after uploading will get below screen



#### Fig 6.1.3: Dataset Uploaded Page

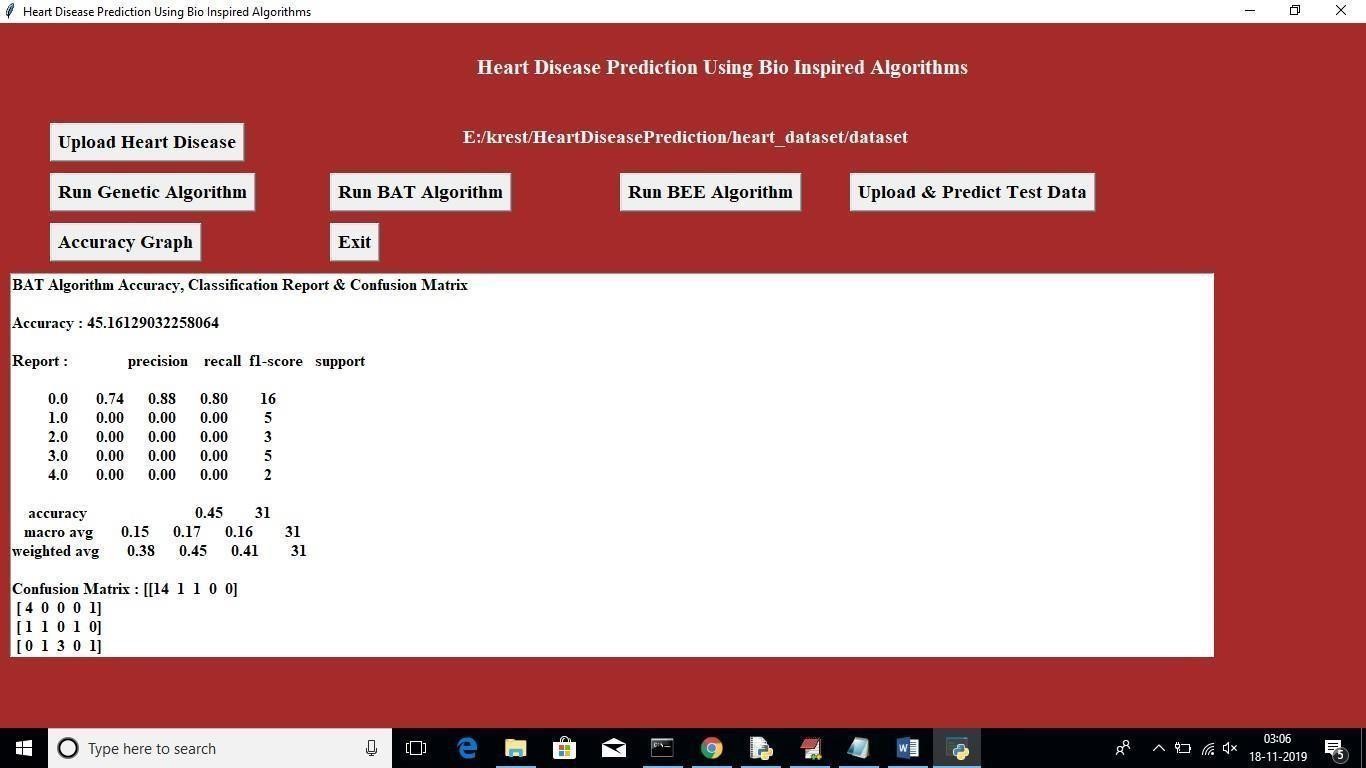
Now click on ‘Run Genetic Algorithm’ button to run genetic algorithm on dataset and to get its accuracy details. While running this algorithm u can see black console to see feature selection process, while running it will open empty windows, u just close all those empty windows except current window



#### Fig 6.1.4 Run Genetic Algorithm

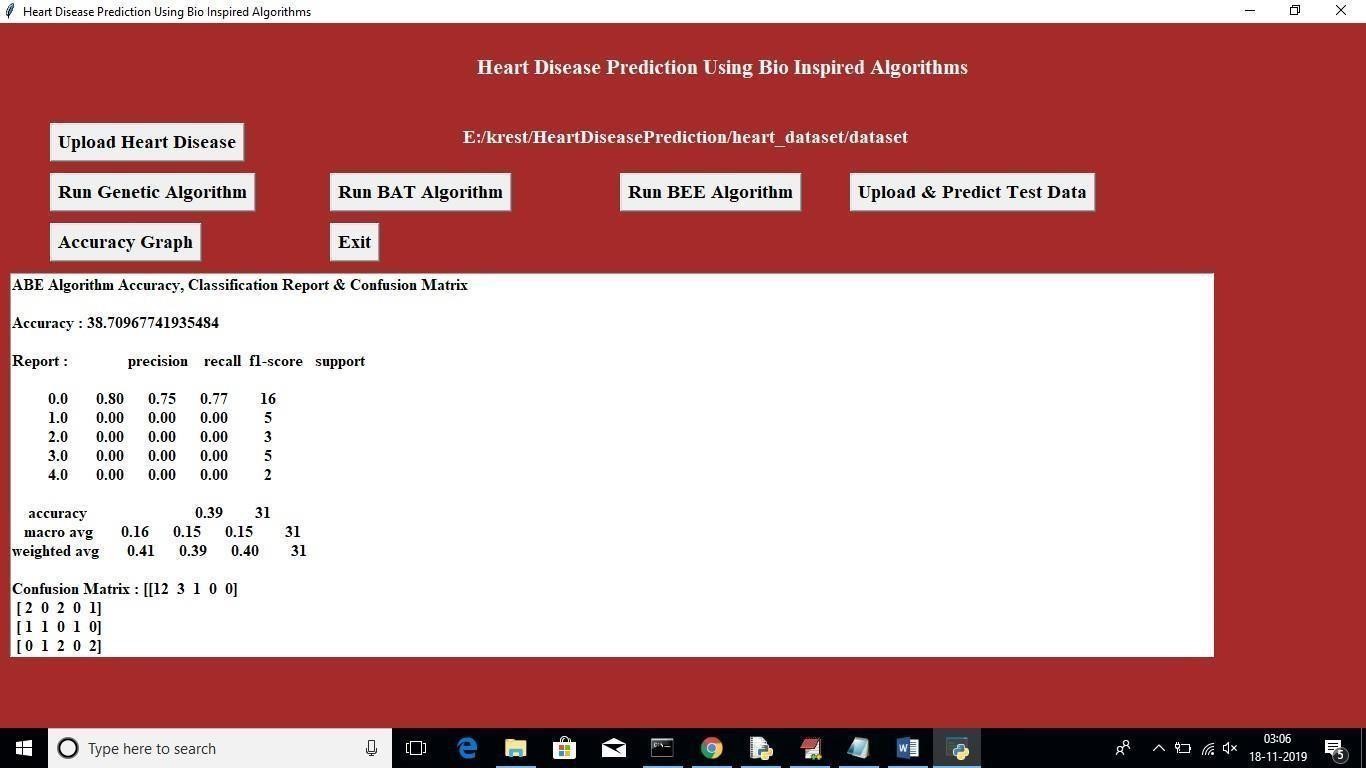
In above screen for GA accuracy, precision and recall we got 100% result. Now click on

‘Run Bat’ algorithm button to get its accuracy



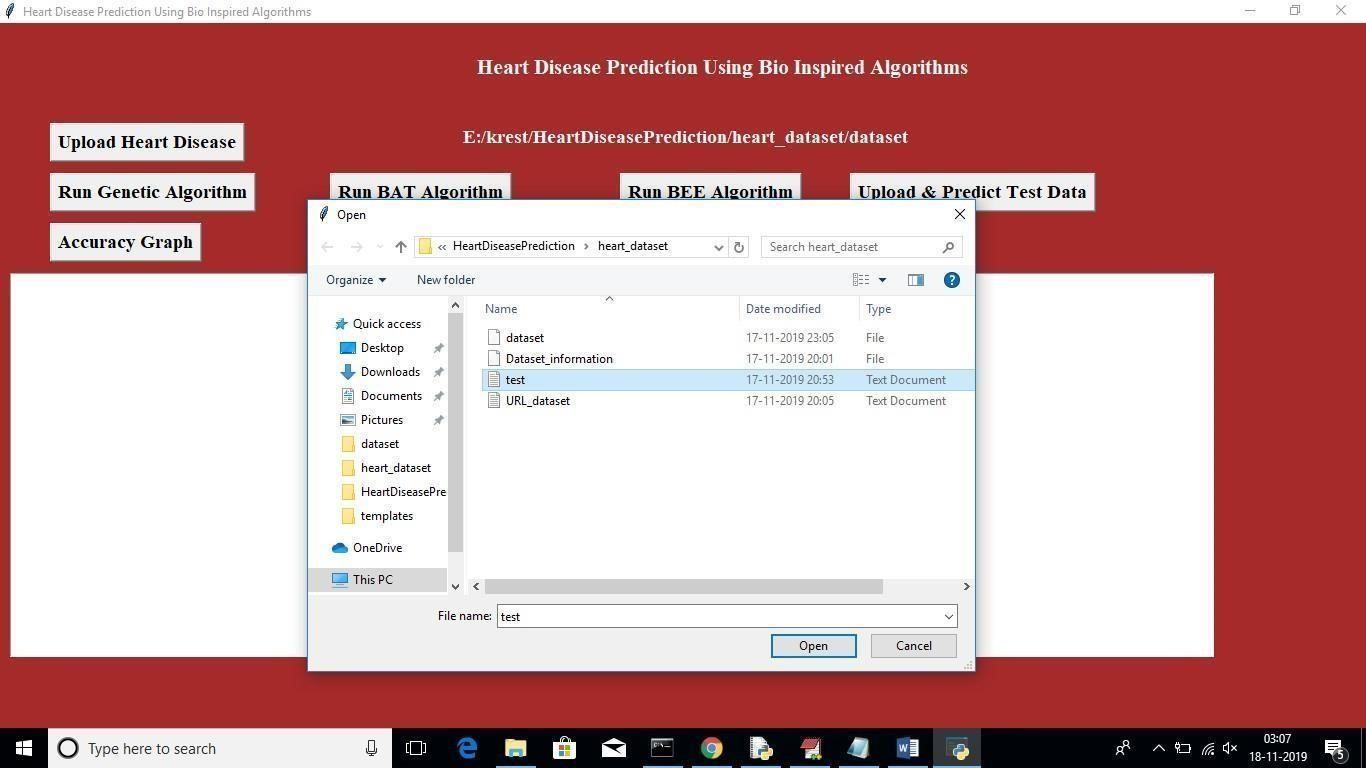
#### Fig 6.1.5: Run BAT Algorithm

In above screen for BAT we got 45% accuracy, now click on ‘Run BEE Algorithm’ button to get BEE accuracy



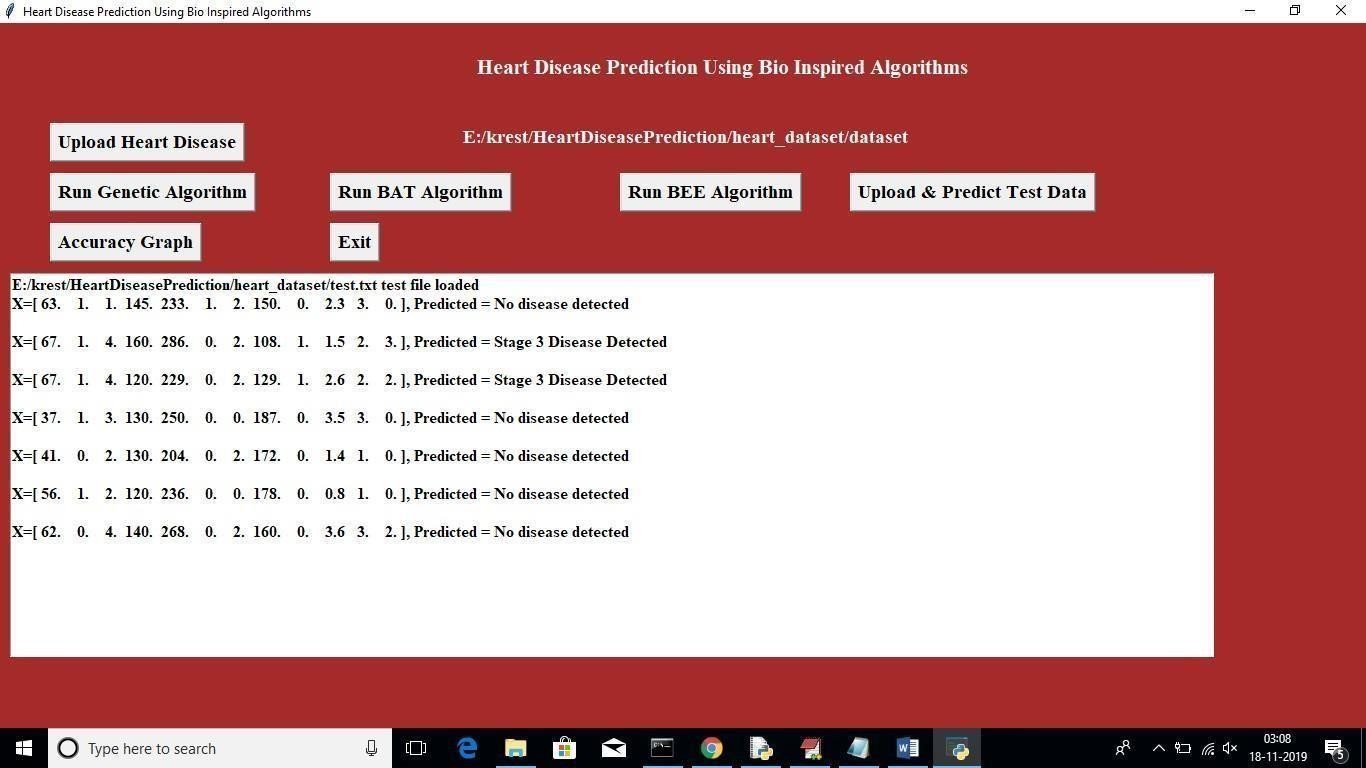
#### Fig 6.1.6: Run BEE Algorithm

In above screen for BEE we got 38% accuracy, now click on ‘Upload & Predict Test Data’ button to upload test data and to predict it class



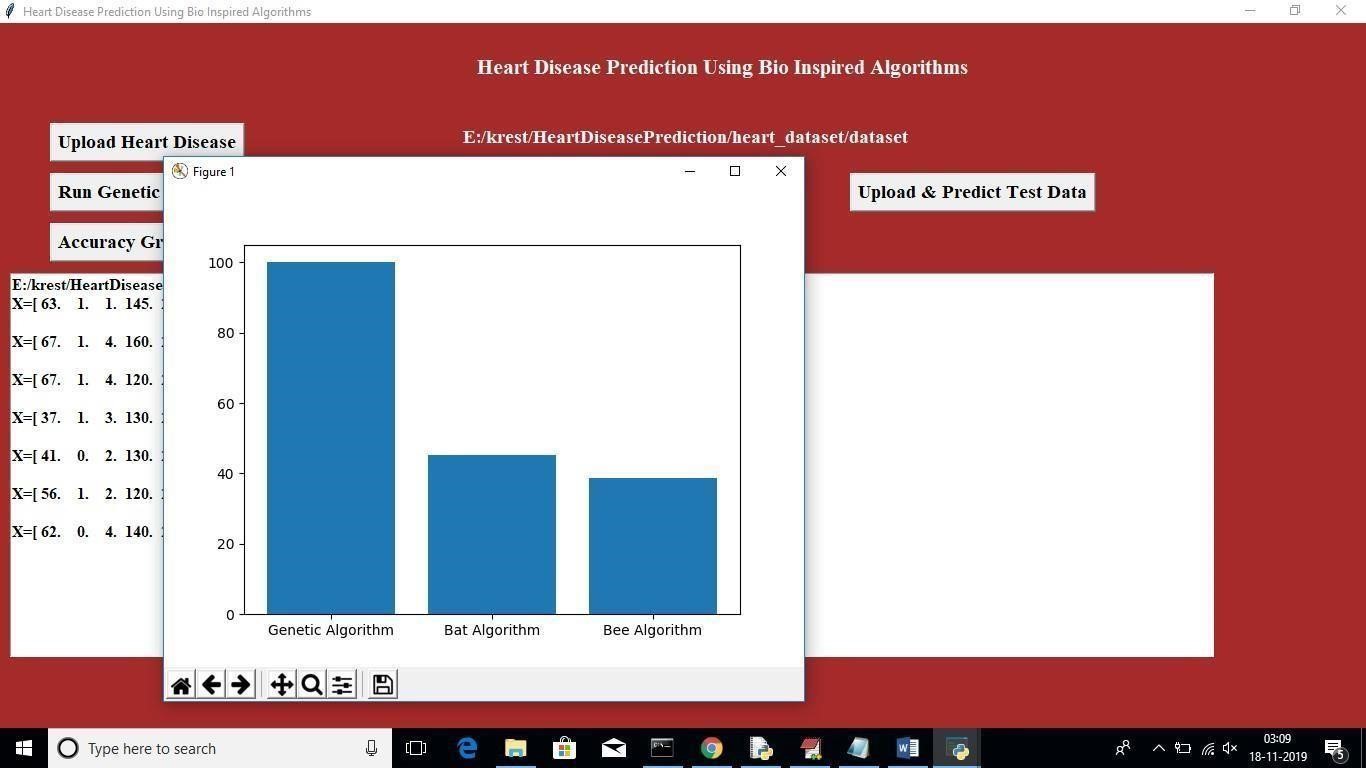
#### Fig 6.1.7: Upload Test File

In above screen I am uploading test file which contains test data without class label, after uploading test data will get below screen



#### Fig 6.1.8: Predicted Disease Stages

In above screen application has predicted disease stages. Now click on ‘Accuracy Graph’ button to view accuracy of all algorithms in graph format



#### Fig 6.1.9: Accuracy Graph

In above graph x-axis represents Algorithm Name and y-axis represents accuracy of those algorithms

## CHAPTER 6

**TESTING & VALIDATION**

### 6.1 INTRODUCTION

In this project student want to detect heart disease from dataset using Bio Inspired features optimizing algorithms such as Genetic Algorithm, Bat and Bee. Here BEE and BAT algorithm is design in python to solve Travelling Salesman Problem to find shortest path and it cannot be implemented with heart disease dataset, so I am implementing 3 algorithms called Genetic, Bat and BEE Algorithm.

Bio inspired algorithms design to optimized features used in dataset for training classification algorithms to increase prediction accuracy, sometime some datasets may have irrelevant values inside dataset and those irrelevant attributes or values may degrade classification accuracy so using optimize algorithms we can reduce features (attribute values) from dataset. This optimize algorithms will be applied on dataset to check whether all values are related to dataset or not, if any attribute found unrelated then it will removed from dataset.

**6.2 TEST CASES AND SCENARIOS:**

1. **Data Collection:**

The first step for predicting the accuracy is data collection and deciding the training and testing dataset. The dataset is taken from the Kaggle website. In this project, we have used 70% training dataset and 30% testing dataset.

1. **Attribute Selection:**

Attribute of dataset are property of dataset which are used for system and for heart many attributes are used such as age of the person, gender of the person, heart bit rate, chest pain type and many more and also predicted output is specified in terms of 0 and 1.

1. **Data Preprocessing:**

To work with categorical variables, we should break each categorical column into dummy columns with 1s and 0s. This step is one the most important step that is to be performed to get accurate result.

1. **Data Balancing:**

Through data balancing we can ensure that both the output classes are balanced to move to further steps. “0” represents that the person is predicted with heart disease and “1” represents that the person is predicted without heart disease.

1. **Histogram of attributes:**

Histogram helps in understanding of each attribute clearly. The best part about this type of plot is that it just takes a single command to draw the plots and it provides so much information in return.

### 6.3 VALIDATION

The validation of these project provides deep insight into machine learning techniques for the classification of heart diseases. The role of a classifier is crucial in the healthcare industry so that the results can be used for predicting the treatment which can be provided to patients. With the increasing number of deaths due to heart diseases, it has become mandatory to develop a system to predict heart diseases effectively and accurately. The motivation for the study was to find the most efficient ML algorithm for the detection of heart diseases.

The accuracy of the algorithms in ML is dependent upon the dataset used for training and testing purposes. when we perform the analysis of algorithms on the basis of the dataset. In the future, more machine learning approaches will be used for the best analysis of heartdiseases. We can also combine two or more algorithms to form a hybrid model to get more accuracy in the future.

we propose a new convolutional neural network based multimodal disease risk prediction Bio Inspired algorithms using structured and unstructured data from hospital. To the best of our knowledge, none of the existing work focused on both data types in the area of medical big data analytics. Compared to several typical prediction algorithms, the prediction accuracy of our proposed algorithm reaches 94.8% with a convergence speed which is faster than that of the CNN-based disease risk prediction algorithm.

**CHAPTER 7**

**CONCLUSION**

### CONCLUSION

Heart Disease Prediction has been developed using three bio-inspired algorithm

modelling techniques. This project predicts person with heart disease by extracting the patient medical history that leads to a fatal heart disease from a dataset that includes patients’ medical history such as chest pain, sugar level, blood pressure, serum cholesterol, maximum heart rate achieved etc. This Heart Disease detection system assists a patient based on his/her clinical information of them been diagnosed with a previous heart disease. The algorithms used in this building the given model are Genetic, BAT and BEE algorithms. The accuracy of our model is 100%. Use of more training data ensures the higher chances of the model to accurately predict whether the given person has a heart disease or not. By using these, computer aided techniques we can predict the patient fast and betterand the cost can be reduced very much. There are a number of medical databases that we can work on as these bio-inspired techniques are better and they can predict better than a human being which helps the patient as well as the doctors. Therefore, in conclusion this project helpsus predict the patients who are diagnosed with heart diseases by cleaning the dataset and applying genetic algorithm. Also, it is concluded that accuracy of BAT algorithm is 45% accuracy and the BEE algorithm is of 38% accuracy. Finally Genetic algorithm will give us 100% accuracy. From this we conclude that Genetic algorithm will give high accuracy than BAT and BEE algorithms. This also provided a view accuracy of all algorithms in graph format. In graph xaxis represents Algorithm Name and y-axis represents Accuracy of those algorithms.

### REFERENCES

1. M. Chen, S. Mao, and Y. Liu, ‘‘Big data: A survey,’’ Mobile Netw. Appl., vol. 19, no.2, pp. 171–209, Apr. 2014.

1. J. Wang, M. Qiu, and B. Guo, ‘‘Enabling real-time information service on telehealth system over cloud-based big data platform,’’ J. Syst. Archit., vol. 72, pp. 69–79, Jan. 2017.
2. D. W. Bates, S. Saria, L. Ohno-Machado, A. Shah, and G. Escobar, ‘‘Big data in health care: Using analytics to identify and manage high-risk and high-cost patients,’’ Health Affairs,vol. 33, no. 7, pp. 1123–1131, 2014.
3. D. Tian, J. Zhou, Y. Wang, Y. Lu, H. Xia, and Z. Yi, ‘‘A dynamic and self-adaptive network selection method for multimode communications in heterogeneous vehicular telematics,’’ IEEE Trans. Intell. Transp. Syst., vol. 16, no. 6, pp. 3033–3049, Dec. 2015.
4. K.Prasanna Lakshmi, Dr. C.R.K.Reddy, “Fast Rule-Based Heart Disease Prediction using Associative Classification Mining”, IEEE International Conference on Computer, Communication and Control (IC4-2015).

1. Boshra Bahrami, Mirsaeid Hosseini Shirvani, “Prediction and Diagnosis of Heart

Disease by Data Mining Techniques”, Journal of Multidisciplinary Engineering Science and Technology (JMEST) ISSN: 3159-0040 Vol. 2 Issue 2, February–2015.

1. N. Nori, H. Kashima, K. Yamashita, H. Ikai, and Y. Imanaka, ‘‘Simultaneous modelingof multiple diseases for mortality prediction in acute hospital care,’’ in Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2015, pp. 855–864.

1. M.Satish, D Sridhar, “Prediction of Heart Disease in Data Mining Technique”, International Journal of Computer Trends & Technology (IJCTT), 2015.
2. Lokanath Sarangi, Mihir Narayan Mohanty, Srikanta Pattnaik, “An Intelligent Decision Support System for Cardiac Disease Detection”, IJCTA, International Press 2015.
3. S. Bandyopadhyay et al., ‘‘Data mining for censored time-to-event data: A Bayesian network model for predicting cardiovascular risk from electronic health record data,’’ Data Mining Knowl. Discovery, vol. 29, no. 4, pp. 1033–1069, 2015.
4. B. Qian, X. Wang, N. Cao, H. Li, and Y.-G. Jiang, ‘‘A relative similarity based method for interactive patient risk prediction,’’ Data Mining Knowl. Discovery, vol. 29, no. 4, pp. 1070–1093, 2015.