IJARST
International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

HEARTDISEASE PREDICTION USING BIOINSPIRED ALGORITHMS

[1,2,3,4] **Sonthe Srivandana, Guggilla Akhil, Gattumeedi Rupa Sri, Morthad Sanjay Kumar,** [5]**V. Anil Kumar**

[1,2,3,4] Ug scholars, MallaReddy college Of Engineering , Hyderabad - 500100
[5] Assistant Professor, MallaReddy college Of Engineering , Hyderabad - 500100

ABSTRACT

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 analyses 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 INTRODUCTION

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 fulfil the progressions in the illness and its affecting elements. With the improvement of enormous information examination innovation, more consideration

CHAPTER - 2

LITERATURE SURVEY

2.1 LITERATURE SURVEY

Literature survey 1 :-

Title :- Effective heart disease prediction using hybrid machine learning techniques

Year :- 2019

Authors:- Mohan, S., Thirumalai, C., & Srivastava, G.

Abstract: The numerous research methodologies taken into account in this work for the categorization and prediction of heart disease using ML and DL are extremely accurate in determining the effectiveness of these methods.According to the findings of this article, ANN has been found to provide the highest accuracy when compared to earlier approaches.Heart illness is predicted using the back propagation multilayer perceptron (MLP) of ANN. NN, DT, SVM, and Naïve Bayes are utilised to find patterns in the data of heart disease patients collected from the UCI laboratory.

Literature survey 2:-

Title:- Recursion enhanced random forest with an improved linear model (RERF-ILM) for heart disease detection on the internet of medical things platform.

Year:- 2020

Authors:- Guo, C., Zhang, J., Liu, Y., Xie, Y., Han, Z., & Yu, J.

Abstract:-

. Heart disease is a group of illnesses that affect people's hearts and veins. Depending on the precise form of cardiac disease, there are various symptoms. Numerous things affect this, including ageing, diabetes, smoking, being overweight, eating junk food, and so forth. Numerous factors that either cause or aggravate heart disease have been found.

CHAPTER - 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

Heart diseases have emerged as one of the most prominent cause of death all around the world. According to World Health Organisation, heart related diseases are responsible for the taking 17.7 million lives every year, 31% of all global deaths. In India too, heart related diseases have become the leading cause of mortality [1]. Heart diseases have killed 1.7 million Indians in 2016, according to the 2016 Global Burden of Disease Report, released on September 15,2017.

3.2 DRAWBACKS

1. Limited complexity for capturing intricate relationships.
2. Dependence on manual feature engineering.
3. Sensitivity to data quality and biases.
4. Difficulty handling high-dimensional data efficiently.

3.3 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.

3.4 ADVANTAGES

1. **Complex Pattern Recognition:** Bioinspired algorithms excel at capturing intricate, non-linear patterns in heart disease data.
2. **Adaptability:** These algorithms can adapt and learn from new data, making them suitable for dynamic healthcare environments.

3.5 SYSTEM REQUIREMENTS

Hardware:

Main Processor	: 2GHz
Ram	: 512 MB (min)
Hard Disk	: 80 GB

Software:

OS : Windows or Linux
Python IDE : python 2.7.x and above

Pycharm IDE Required,jupyter notebook.

Setup tools and pip to be installed for 3.6 and above

Language Python Scripting

3.6 FEASIBILITY STUDY

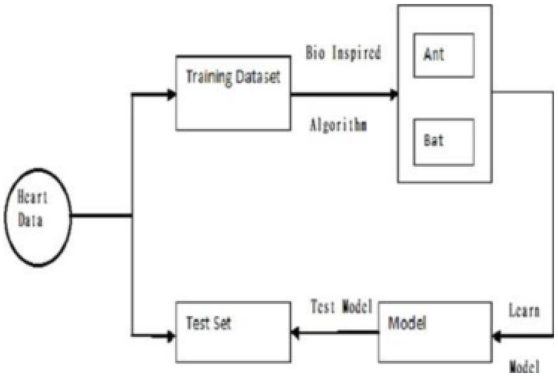
Understanding the Problem

Start by comprehensively understanding heart disease, its risk factors, available data sources (like patient records, test results), and the significance of accurate prediction for early intervention.

CHAPTER - 4

SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE



System Architecture

Level 0:

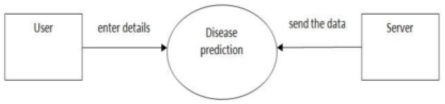
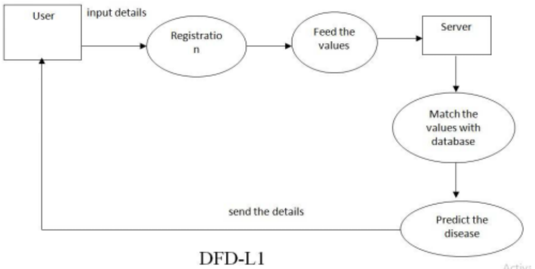


Fig4.3.1

Level 1:



DFD-L1

Fig4.3.2

4.2 MODULES

1. Upload Training Data
2. Data Pre-Processing

1. UPLOAD TRAINING DATA

The process of rule generation advances in two stages. During the first stage, the SVM model is built using training data During each fold, this model is utilized for predicting the class labels.

2. DATA PREPROCESSING

Heart disease data is pre-processed after collection of various records. The dataset contains a total of 303 patient records, where 6 records are with some missing values . Those 6 records

4.3 UML DIAGRAMS

DATA FLOW DIAGRAM:

A data flow diagram (DFD) is a graphical representation of the “flow” of data through an information system.

SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams

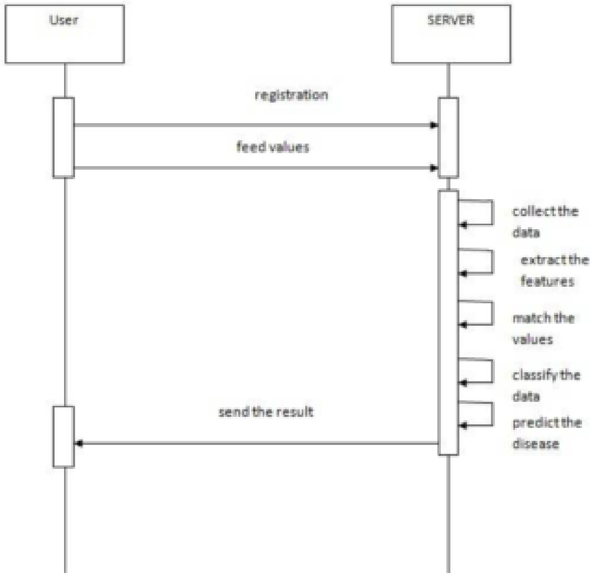


Fig4.3.5. Sequence diagram

CHAPTER - 5

SYSTEM IMPLEMENTATION

5.1 Machine Learning

Machine Learning is a system that can learn from examples through self-improvement and without being explicitly coded by the programmer. The breakthrough comes with the idea that a machine can singularly learn from the data (i.e., an example) to produce accurate results.

5.1.1 Working of Machine Learning

Machine learning is the brain where all the learning takes place. The way the machine learns is similar to the human being. Humans learn from experience. The more we know, the more easily we can predict. By analogy, when we face an unknown situation, the likelihood of success is lower than the known situation.

5.1 PYTHON

Python programming language is used for building the machine learning model.

5.2.1 Introduction

Python is an object-oriented, high level language, interpreted, dynamic and multipurpose programming language. Python is easy to learn yet powerful and versatile scripting language which makes it attractive for Application Development. Python's syntax and dynamic typing with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas. Python supports multiple programming pattern, including object oriented programming, imperative and functional programming or procedural styles.

5.3 FUNDAMENTALS OF PYTHON

This section contains the basic fundamentals of Python.

5.3.1 Tokens

Tokens can be defined as a punctuator mark, reserved words and each individual word in a statement. Token is the smallest unit inside the given program. Tokens include Keywords,

Identifiers, literals, operators.

5.3.2 Tuples

Tuple is another form of collection where different type of data can be stored. It is similar to list where data is separated by commas. Only the difference is that list uses square bracket and tuple uses parenthesis. Tuples are enclosed in parenthesis and cannot be changed.

Eg: tuple=('rahul',100,60.4,'deepak')

5.4 JUPYTER NOTEBOOK

Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. It provides an interactive computing environment that supports various programming languages, including Python, R, Julia, and more. Here are some key notes on Jupyter Notebook:

5.5VISUAL STUDIO CODE

Visual Studio Code (VS Code) is a free and lightweight source code editor developed by Microsoft. It is widely used by developers for various programming languages and platforms. Visual Studio Code (VS Code) can be used in several ways:

5.6 Algorithm

5.6.1 Genetic Algorithm

- A genetic algorithm (GA) is a heuristic optimization technique inspired by the process of natural selection and evolution. It's used to find approximate solutions to complex optimization and search problems. GAs are particularly useful for solving problems where traditional methods may be impractical due to their high dimensionality, non-linearity, or other challenging characteristics.

5.6.2 Bat Algorithm

- The Bat Algorithm is known for its ability to explore complex search spaces and its adaptability to different types of optimization problems. Its parameters, such as loudness and pulse rate, play a crucial role in balancing exploration and exploitation. The algorithm has been applied to a variety of optimization problems in different fields, including engineering, economics, and science.

5.8 SOURCE CODE

```
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 from __future__ import
print_function
from tkinter
 import message box
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
```

CHAPTER - 6

TESTING

6.1 TESTING

6.1.1 White Box Testing

White-box testing (also known as clear box testing, glass box testing, transparent box testing, and structural testing) is a method of testing software that tests internal structures or workings of an application, as opposed to its functionality (i.e. black-box testing).

- Control flow testing
- Data flow testing
- Branch testing

6.1.2 Black Box Testing

Black-box testing is a method of software testing that examines the functionality of an application (e.g. what the software does) without peering into its internal structures or workings (see white-box testing). This method of test can be applied to virtually every level of software testing: unit, integration,

system and acceptance. It typically comprises most if not all higher-level testing, but can also dominate unit testing as well

Test procedures

6.2 TESTING METHODS

Train-Test Split

Divide your dataset into training and testing subsets. The common split is 70-30 or 80-20 for training and testing, respectively.

CHAPTER – 7

RESULTS

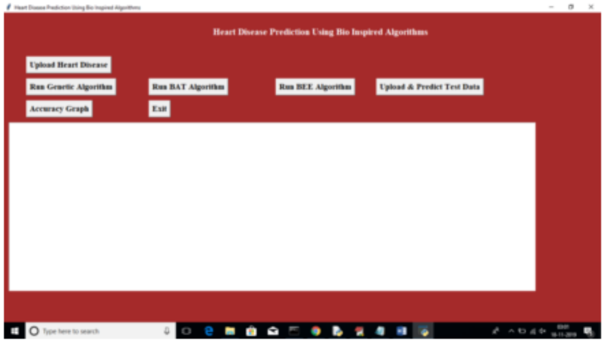


Fig 7.1 Output Screen

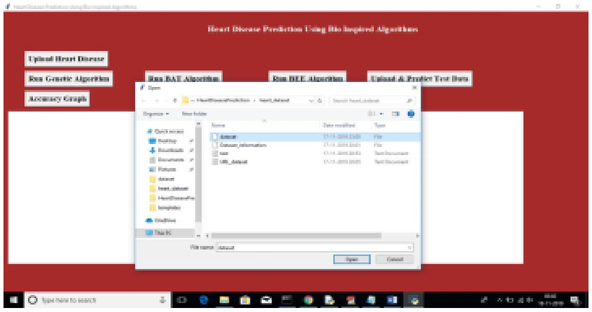
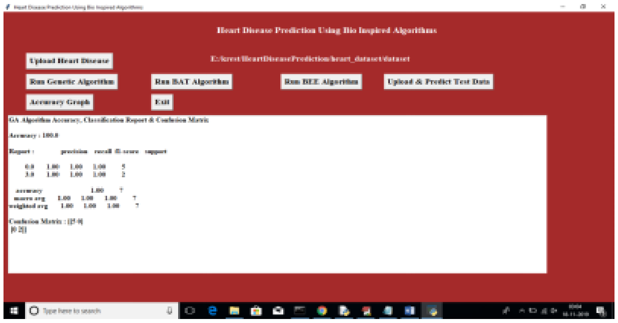


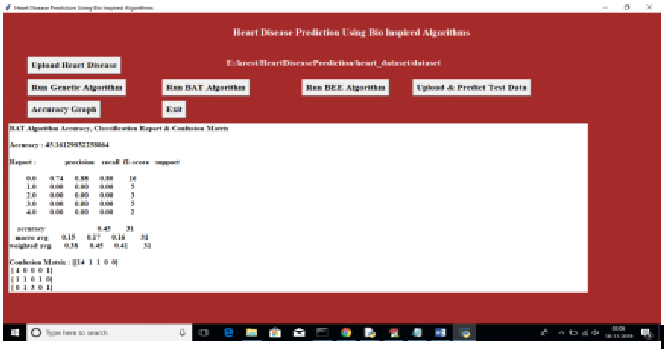
Fig 7.2 Uploading dataset



E:/kresi/HeartDiseasePrediction/heart_dataset/dataset

```
GA Algorithm Accuracy, Classification Report & Confusion Matrix

Accuracy : 100.0

Report :           precision   recall  f1-score   support

       0.0      1.00      1.00      1.00         5
       3.0      1.00      1.00      1.00         2

    accuracy                          1.00         7
   macro avg      1.00      1.00      1.00         7
weighted avg      1.00      1.00      1.00         7

Confusion Matrix : [[5 0]
 [0 2]]
```

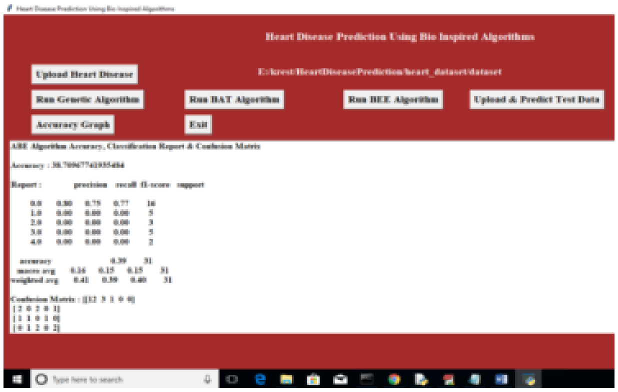
Fig 7.3 Genetic Algorithm



E:/kresi/HeartDiseasePrediction/heart_dataset/dataset

```
BAT Algorithm Accuracy, Classification Report & Confusion Matrix

Accuracy : 45.16129032258064

Report :           precision   recall  f1-score   support

       0.0      0.74      0.88      0.80        16
       1.0      0.00      0.00      0.00         5
       2.0      0.00      0.00      0.00         3
       3.0      0.00      0.00      0.00         5
       4.0      0.00      0.00      0.00         2

    accuracy                          0.45        31
   macro avg      0.15      0.17      0.16        31
weighted avg      0.38      0.45      0.41        31

Confusion Matrix : [[14  1  1  0  0]
 [ 4  0  0  0  1]
 [ 1  1  0  1  0]
 [ 0  1  3  0  1]
```

Fig 7.4 Bat Algorithm



Heart Disease Prediction Using Bio Inspired Algorithms

E:/krest/HeartDiseasePrediction/heart_dataset/dataset

```
ABE Algorithm Accuracy, Classification Report & Confusion Matrix

Accuracy : 38.70967741935484

Report :            precision   recall f1-score   support

     0.0      0.80      0.75      0.77        16
     1.0      0.00      0.00      0.00         5
     2.0      0.00      0.00      0.00         3
     3.0      0.00      0.00      0.00         5
     4.0      0.00      0.00      0.00         2

   accuracy                        0.39        31
  macro avg      0.16      0.15      0.15        31
weighted avg     0.41      0.39      0.40        31

Confusion Matrix : [[12  3  1  0  0]
 [ 2  0  2  0  1]
 [ 1  1  0  1  0]
 [ 0  1  2  0  2]
```

Fig.7.5 Bee Algorithm

CHAPTER - 8

CONCLUSION

8.1 CONCLUSION

Three bio-inspired algorithm modeling methodologies were used to create a method for predicting heart disease. This work extracts the individual's medical information that leads to deadly cardiovascular disease from the dataset that comprises patient medical history, including chest x-rays, hospitalizations, and medications,symptoms, blood sugar, blood pressure, cholesterol levels, maximal heart rate and so forth. If a patient has already been diagnosed with heart disease, this heart disease detection method may help.

CHAPTER - 9

FUTURE ENHANCEMENT

9.1 FUTURE ENHANCEMENTS

Integrating bioinspired algorithms into heart disease prediction systems can enhance their accuracy and efficiency. Bioinspired algorithms are computational methods inspired by natural processes and biological systems. Here are some potential future enhancements for heart disease prediction using bioinspired algorithms:

REFERENCES

1. Soni, Jyoti, Ujma Ansari, Dipesh Sharma, and SunitaSoni. "Predictive data mining for medical diagnosis: An overview of heart disease prediction." International Journal of Computer Applications, vol.17, no. 8, pp.43-48, 2011.

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. LokanathSarangi, Mihir Narayan Mohanty, SrikantaPattnaik, “An Intelligent Decision Support System for Cardiac Disease Detection”, IJCTA, International Press 2015.

4. M. Chen, P. Zhou, and G. Fortino, ‘‘Emotion communication system,’’ IEEE Access, vol. 5, pp. 326337,2017,doi:10.1109/ACCESS.2016.2641480

5. J. Wang, M. Qiu, and B. Guo, ‘‘Enabling real-time information service on telehealth system over cloud-based bigdata platform,’’J.Syst.Archit.vol.72,pp.69–79, Jan.2017.