A

REPORT

ON

ANALYSIS OF HEART DISEASE OF A PATIENT

BY

KARTIKEY DWIVEDI **2016CSE283**

AMARNATH MISHRA **2016CSE305**

AT

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CSE 219 BIG DATA ANALYTICS

MINI PROJECT REPORT

PRESIDENCY UNIVERSITY, BENGALURU

DEC 2019

DECLARATION

I hereby declare that the report entitled “Analysis of heart disease ” is submitted to CSE219 big data analytics lab is carried out by us. The material contained in this report has not been submitted to anywhere else.

Name of the student	ID Number	Signature
KARTIKEY DWIVEDI	**2016CSE283**	
AMARNATH MISHRA	**2016CSE305**	

ABSTRACT

Analyzing the health data is always been difficult. To overcome this problem, we implement MapReduce programming. The objective of this project is to-

- To identify heart disease because of several contributory risk factors such as diabetes, high blood pressure, high cholesterol, abnormal pulse rate and many other factors.

Dataset contains 6 columns which comprises of different risk factors which helps to predict whether the person is affected from heart disease or not.

Signature of Student: (Kartikey Dwivedi)

Signature of Student: (Amarnath Mishra)

Guided by

Date:

Dr. R.Sathish Kumar

TABLE OF CONTENTS

INTRODUCTION

Big data has shown very promising results in the field of healthcare. So, here we are applying this technology to predict whether the patient is affecting from the heart disease or not. The term “heart disease” is often used interchangeably with the term “cardiovascular disease”.

Cardiovascular disease generally refers to conditions that involve narrowed or blocked blood vessels that can lead to a heart attack, chest pain (angina) or stroke.

Other heart conditions, such as those that affect your heart’s muscle, valves or rhythm, also are considered forms of heart disease. The amount of data in the healthcare industry is huge. According to a us news article heart disease proves to be the leading cause of death for both women and men. The article states following:

About 610,000 people die of heart disease in the United States every year–that’s 1 in every 4 deaths. Heart disease is the leading cause of death for both men and women. More than half of the deaths due to heart disease in 2009 were in men.

Coronary Heart Disease (CHD) is the most common type of heart disease, killing over 370,000 people annually. Every year about 735,000 Americans have a heart attack. Of these, 525,000 are a first heart attack and 210,000 happen in people who have already had a heart attack.

This makes heart disease a major concern to be dealt with. But it is difficult to identify heart disease because of several contributory risk factors such as diabetes, high blood pressure, high cholesterol, abnormal pulse rate and many other factors. Due to such constraints we are solving this problem using Hadoop.

OBJECTIVE OF THE PROJECT

The main objective of this project is to do the analysis of the data of the patient record and find which patient can get a heart attack in future based on his present record.

The MapReduce program will divide the patients in three category-

a) Safe in future
b) May get heart attack in future
c) Will get a heart attack in future

As we all know that a person gets heart attack based on several contributory risk such as diabetes, high blood pressure, high cholesterol, abnormal pulse rate and many other factors.

Hence, to analysis such record we have used MapReduce programming to find a weighted sum of all features contributing in heart attack. This final weighted sum is used to find a decision boundary for all three categories. And, later this result in stored in HDFS.

DESIGN AND IMPLEMENTATION

DATASET DESCRIPTION-

To develop such program, we have to first collect a structured data. For our project we have used a dataset of 23 entries. There are 6 columns in the dataset, which are described below.

a. Age: displays the age of the individual.
b. Sex: displays the gender of the individual using the following format: 1 = male 0 = female

c. High Blood Pressure: displays the high blood pressure value of an individual in mmHg (unit)
d. Low Blood Pressure: displays the low blood pressure value of an individual in mmHg (unit)
e. Serum Cholesterol: displays the serum cholesterol in mg/dl (unit)
f. Max heart rate achieved: displays the max heart rate achieved by an individual.
g. Diagnosis of heart disease: Displays whether the individual may get an heart disease or not in future.

MODIFICATION IN DATASET-

Every data entry is modified to a scale of 1-5.

a. SEX- Sex of the patient is taken as –
    i. Male-1 Female-0
b. HIGH BP-High BP values are converted to scale of 1-5 as follows-
    i. 120-129=1 130-139=2
    ii. 140-149=3 150-159=4
    iii. 160+=5
c. LOW BP-Low BP values are converted to scale of 1-5 as follows-
    i. 40-49=5 50-59=4
    ii. 60-69=3 70-79=2
    iii. 80-89=1 90-99=2
    iv. 100-109=3 110-119=4
    v. 120+=5
d. SERUM CHOLESTROL- Serum cholesterol values are converted to the scale of 1-5 as follows-
    i. 190-200=1 201-210=2
    ii. 211-220=3 221-230=4
    iii. 231+=5

e. HEARTRATE- heart rate values are converted to the scale of 1-5 as follows-

1. <50=4 51-60=3
2. 61-70=2 71-80=1
3. 80-89=2 90-99=3
4. 100-110=4 >111=5

DATASET CREATION-

The following dataset in created in Desktop/heartdata.txt file-

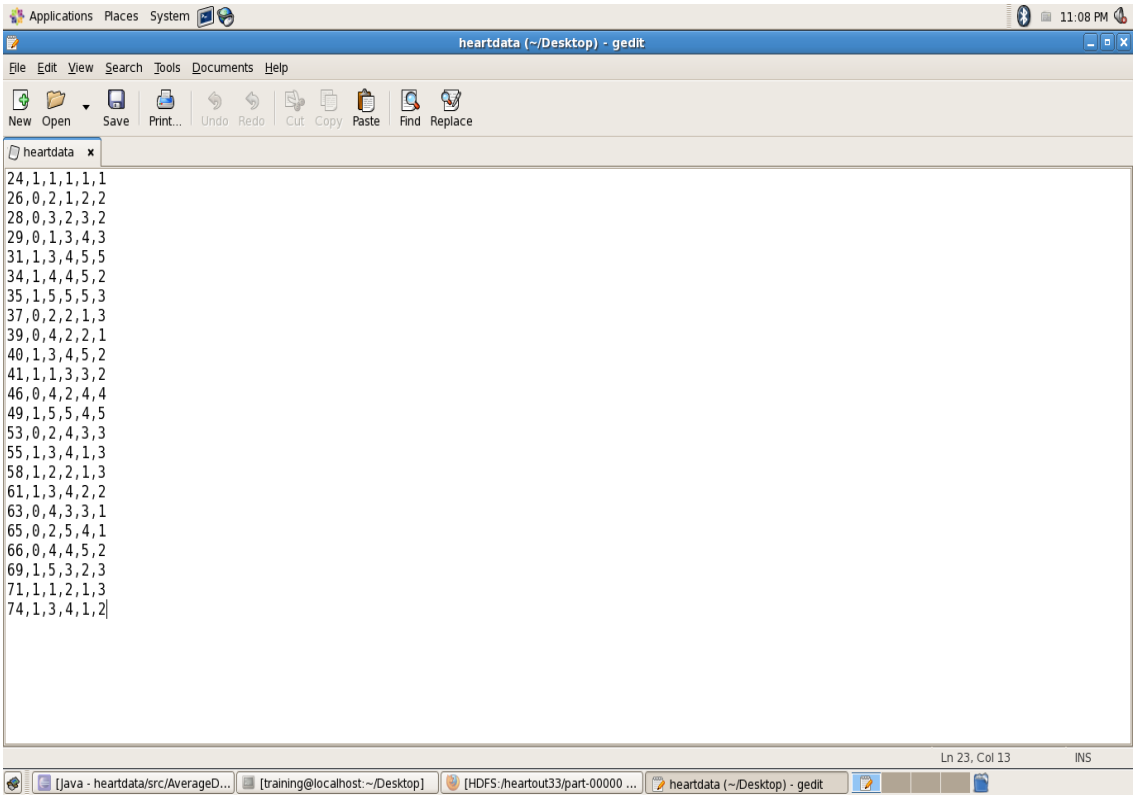

FIGURE 1 Dataset file-heartdata.txt

TRAINING AND PROGRAMMING

To develop this project three files were created-

1. Heartdisease.java
2. AverageDiseaseMapper.java
3. AverageDiseaseReducer.java

Heartdisease.java

```
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.conf.Configured;
import org.apache.hadoop.io.FloatWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.FileInputFormat;
import org.apache.hadoop.mapred.FileOutputFormat;
import org.apache.hadoop.mapred.JobClient;
import org.apache.hadoop.mapred.JobConf;
import org.apache.hadoop.util.Tool;
import org.apache.hadoop.util.ToolRunner;

public class Heartdisease extends Configured implements Tool {

   public int run(String[] args) throws Exception {
      if(args.length<2)
      {
         System.out.println("Plz Give Input Output Directory Correctly");
         return -1;
      }
      JobConf conf = new JobConf(Heartdisease.class);
      FileInputFormat.setInputPaths(conf,new Path(args[0]));
```

```
        FileOutputFormat.setOutputPath(conf, new Path(args[1]));
        conf.setMapperClass(AverageDiseaseMapper.class);
        conf.setReducerClass(AverageDiseaseReducer.class);
        conf.setMapOutputKeyClass(Text.class);
        conf.setMapOutputValueClass(FloatWritable.class);
        conf.setOutputKeyClass(Text.class);
        conf.setOutputValueClass(Text.class);
        JobClient.runJob(conf);
        return 0;
        }

    public static void main(String[] args) throws Exception {
        int exitcode = ToolRunner.run(new Heartdisease(), args);
        System.exit(exitcode);
        }
}
```

AverageDiseaseMapper.java

```
import java.io.IOException;
import org.apache.hadoop.io.FloatWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reporter;
import org.apache.hadoop.mapred.Mapper;

public class AverageDiseaseMapper extends MapReduceBase implements
Mapper<LongWritable, Text, Text, FloatWritable> {

    @Override
    public void map(LongWritable key, Text value,
```

```
        OutputCollector<Text, FloatWritable> output, Reporter r)
            throws IOException {

        String line = value.toString();
        String[] items = line.split(",");

        String age = items[0];
        Float sex=Float.parseFloat(items[1]);
        Float highbp=Float.parseFloat(items[2]);
        Float lowbp=Float.parseFloat(items[3]);
        Float serumcontrol=Float.parseFloat(items[4]);
        Float heartrate=Float.parseFloat(items[5]);
        Float average=(sex+highbp+lowbp+serumcontrol+heartrate)/5;


        output.collect(new Text(age), new FloatWritable(average));

    }
}
```

AverageDiseaseReducer.java

```
import java.io.IOException;
 import java.util.Iterator;

 import org.apache.hadoop.io.FloatWritable;
 import org.apache.hadoop.io.Text;
 import org.apache.hadoop.mapred.MapReduceBase;
 import org.apache.hadoop.mapred.OutputCollector;
 import org.apache.hadoop.mapred.Reducer;
 import org.apache.hadoop.mapred.Reporter;
```

```
public class AverageDiseaseReducer extends MapReduceBase implements
Reducer<Text,FloatWritable,Text,Text>
{
   @Override
   public void reduce(Text key, Iterator<FloatWritable> values,
         OutputCollector<Text, Text> output, Reporter r)
         throws IOException {
      //Iterate all and calculate maximum
     while (values.hasNext()) {
         FloatWritable i = values.next();
         float value=i.get();
         if(value>0.8 && value<=1.5){
        output.collect(key, new Text("Safe in future"));
         }
         else if(value>1.6 && value<=2.3)
         {
        output.collect(key, new Text("May get heart attack in future"));
         }
         else
         {
        output.collect(key, new Text("Will get a heart attack in future"));
        }
      }
   }
}
```

RESULT AND ANALYSIS

To analyze the data, we type following code in terminal-

- hadoop dfs -put heartdata /heartout3
- hadoop jar heart.jar Heartdisease /heartout3.txt /heartout33

To view the output in browser, we type-

- Hadoop Administrator ⟶ heartout33 ⟶ part-00000

Output contains two columns as follows-

- AGE- shows the age of the patient
- CATEGORY- shows which category the patient belongs

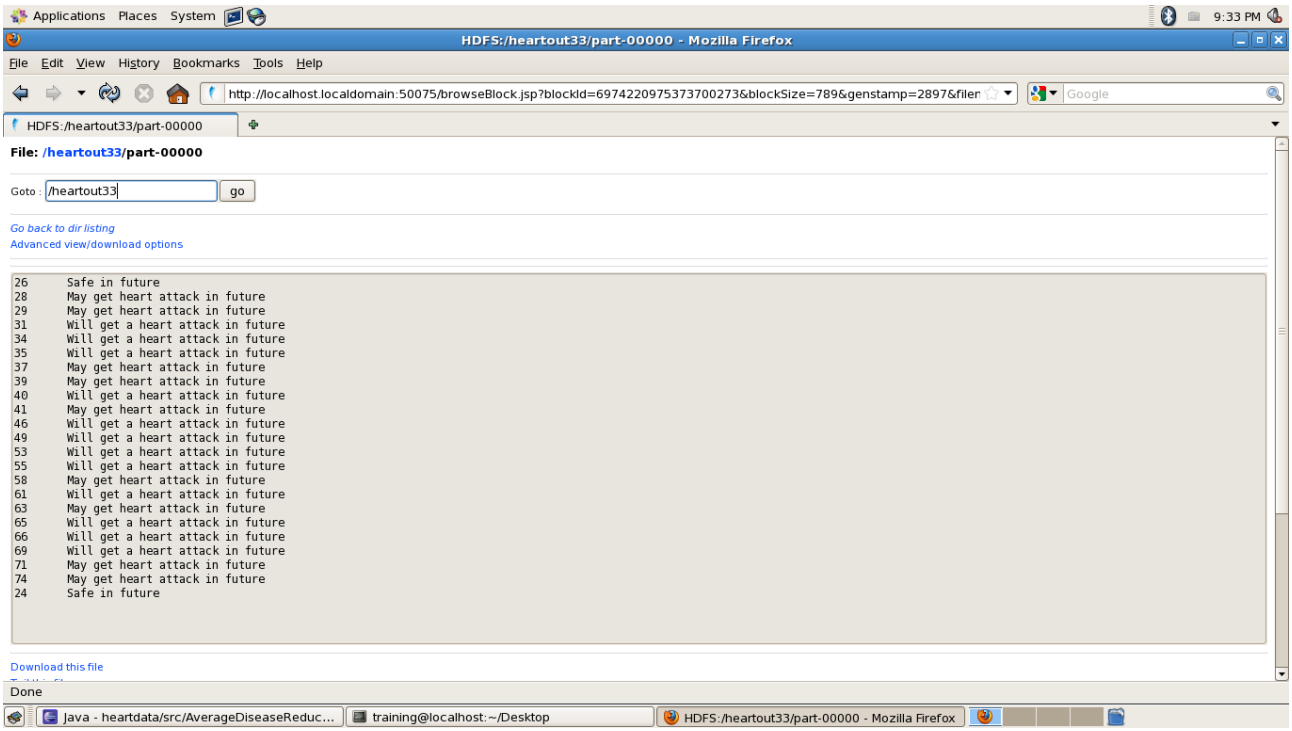

FIGURE-2 OUTPUT OF THE PROJECT STORED IN HDFS

CONCLUSION AND FUTURE ENHANCEMENTS

- The project aims to analyze the data and tells to which category does the patient belongs.

- The project is able to successfully identify the category of the patient for large dataset. It can be used as a reference during the treatment of the patient for analyzing his current condition.

For, future enhancement we can add additional risk factors in dataset like chest-pain type, fasting Blood sugar, Thal, Number of major vessels (0–3) colored by fluoroscopy etc. to analyze the category of patient more accurately. Later, we can also create a website which allows users to enter their health data and find to which category they belong to. This will help user to analyze their health status free of cost and will also help dataset to develop the decision boundary more accurately as the data grows further.

REFERENCES

[1]. To choose the features in our dataset and patient health record https://archive.ics.uci.edu/ml/datasets/Heart+Disease

[2]. Facts regarding heart attack risk factors https://www.heartfoundation.org.au/your-heart/know-your-risks/heart-attack-risk-factors