# INNOVATIVE PROJECT DEVELOPMENT REPORT

**DETECTION OF THYROID DISORDERS USING MACHINE LEARNING APPOARCH**

***Submitted by***

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **P. USHA SRI** | **P. LAXMI SANJANA L. HARSHINI** | **P.YASHASHWINI** |
| **(21RH1A05H5)** | **(21RH1A05J2) (21RH1A05D1)** | **(21RH1A05H3)** |

***Under the Esteemed Guidance of***

**Mrs. K. RAMYA SRI**

**Assistant Professor**

***In partial fulfilment of the Academic Requirements for the Degree of***

**BACHELOR OF TECHNOLOGY**

# Computer Science & Engineering

****

**MALLA REDDY ENGINEERING COLLEGE FOR WOMEN**

(**Autonomous Institution-UGC, Govt. of India**)

***Accredited by NBA & NAAC with ‘A’ Grade, UGC, Govt. of India***

**NIRF Indian Ranking–2018, Accepted by MHRD, Govt. of India Permanently Affiliated to JNTUH,**

**Approved by AICTE, ISO 9001:2015 Certified Institution**

**AAAA+ Rated by Digital Learning Magazine, AAA+ Rated by Careers 360 Magazine,**

**6th Rank CSR, Platinum Rated by AICTE-CII Survey, Top 100 Rank band by ARIIA, MHRD, Govt. of India**

**National Ranking-Top 100 Rank band by Outlook, National Ranking-Top 100 Rank band by Times News Magazine**

**Maisammaguda, Dhullapally, Secunderabad, Kompally-500100**

**2023-2024**

## MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(**Autonomous Institution-UGC, Govt. of India**)

***Accredited by NBA & NAAC with ‘A’ Grade, UGC, Govt. of India***

**NIRF Indian Ranking–2018, Accepted by MHRD, Govt. of India Permanently Affiliated to JNTUH,**

**Approved by AICTE, ISO 9001:2015 Certified Institution**

**AAAA+ Rated by Digital Learning Magazine, AAA+ Rated by Careers 360 Magazine,**

**6th Rank CSR, Platinum Rated by AICTE-CII Survey, Top 100 Rank band by ARIIA, MHRD, Govt. of India**

**National Ranking-Top 100 Rank band by Outlook, National Ranking-Top 100 Rank band by Times News Magazine Maisammaguda, Dhullapally, Secunderabad, Kompally-500100**

***DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING***

### CERTIFICATE

This is to certify that the Project work entitled “**DETECTION OF THYROID DISORDERS USING MACHINE LEARNING APPOARCH”** is carried out by

**P. USHA SRI (21RH1A05H5) P. LAXMI SANJANA (21RH1A05J2)**

**L.HARSHINI (21RH1A05D1) P. YASHASHWINI((21RH1A05H3)**

in partial fulfilment for the award of degree of **BACHELOR OF TECHNOLOGY**

in Computer Science and Engineering, Malla Reddy Engineering Collage for Women (Autonomous), Hyderabad during the academic year 2023-2024.

|  |  |
| --- | --- |
| **Supervisor’s Signature** | **Head of the Department** |
| **Mrs. K. Ramya Sri** | **Dr. Geetha Reddy** |
| Assistant Professor | Professor and HOD |

**EXTERNAL EXAMINER**

## ACKNOWLEDGEMENT

We feel ourselves honoured and privileged to place our warm salutation to our college

**Malla Reddy Engineering College for Women** and Department of **Computer Science and Engineering** which gave us the opportunity to have expertise in engineering and profound technical knowledge.

We wish to convey gratitude to our Principal **Dr. Y. Madhavee Latha**, for providing us with the environment and mean to enrich our skills and motivating us in our endeavour and helping us to realize our full potential.

We would like to thank Prof. **A. Radha Rani**, Director of Computer Science and Engineering & Information Technology for encouraging us to take up a project on this subject and motivating us towards the Project Work.

We express my sincere gratitude to **Dr. Geetha Reddy** Head of the Department of Computer Science and Engineering for inspiring us to take up a project to this subject and successfully guiding us towards its completion.

We would like to thank our internal guide **Mrs.K. Ramya Sri**, Assistant Professor and all the Faculty members for their valuable guidance and encouragement towards the completion of our project work.

**With regards and gratitude,**

**P. USHA SRI (21RH1A05H5)**

**P. LAXMI SANJANA (21RH1A05J2)**

**L.HARSHINI (21RH1A05D1)**

**P. YASHASHWINI (21RH1A05H3)**

## DECLARATION

We hereby declare that our project entitled **“DETECTION OF THYROID DISORDERS USING MACHINE LEARNING APPOARCH”** submitted to **Malla Reddy Engineering College for Women, Hyderabad** for the award of the Degree of Bachelor of Technology in **Computer Science and Engineering** is a result of original research work done by us.

It is declared that the project report or any part thereof has not been previously submitted to any University or Institute for the award of Degree.

**P. USHA SRI (21RH1A05H5)**

**P. LAXMI SANJANA (21RH1A05J2)**

**L.HARSHINI (21RH1A05D1)**

**P. YASHASHWINI (21RH1A05H3)**

### ABSTRACT

Classification based Machine learning plays a major role in various medical services. In medical field, the salient and demanding task is to diagnose patient’s health conditions and to provide proper care and treatment of the disease at the initial stage. Let us consider Thyroid disease as the example. The normal and traditional methods of thyroid diagnosis involve a thorough inspection and also various blood tests. The main goal is to recognize the disease at the early stages with a very high correctness. Machine learning techniques play a major role in medical field for making a correct decision, proper disease diagnosis and also saves cost and time of the patient. The purpose of this study is prediction of thyroid disease using classification Predictive Modelling followed by binary classification using Decision Tree ID3 and Naive Bayes Algorithms. The Thyroid Patient dataset with proper attributes is fetched and using the Decision Tree algorithm the presence of thyroid in the patient is tested. Further, if thyroid is present then Naïve Bayes algorithm is applied to check for the thyroid stage in the patient.

### INDEX

|  |  |
| --- | --- |
| **TITLE** | **PAGE NO.** |
| **1.INTRODUCTION** | **1** |
| Introduction about the project | 1 |
| Existing System | 2 |
| Proposed System | 2 |
| **2.SYSTEM STUDY** | **3** |
| Economic Feasibility | 3 |
| Technical Feasibility | 3 |
| **3.REQUIREMENT ANALYSIS** | **5** |
| Requirement Specification | 5 |
| Languages used to Develop | 5 |
| **4.SYSTEM DESIGN** | **6** |
| System Architecture | 6 |
| Use case Diagram | 6 |
| Class Diagram | 7 |
| Collaboration Diagram | 7 |
| Sequence Diagram | 7 |
| **5.MODULES** | **8** |
| **6. ALGORITHMS** | **10** |
| Navy Baias Algorithm | 10 |
| Support Vector Machine | 11 |
| RANDOM FOREST ALGORITHM | 11 |
| **7.Source Code** | 12 |
| **8.SYSTEM TEST** | **18** |
| Types of Tests | 18 |
| **9.INPUT AND OUTPUT DESIGN** | **21** |
| Input Design | 21 |
| Output Design | 21 |
| Output Screens | 22 |
| **10.CONCLUSION** | **27** |
| **11.REFERENCE** | **28** |

#### CHAPTER 1

**INTRODUCTION**

**INTRODUCTION ABOUT THE PROJECT**

Thyroid disease diagnosis is not a simple task. It involves many procedures. The normal traditional way includes a proper medical examination and many blood samples for blood tests. Therefore, there is a necessity for a model which detects the thyroid disease at a very early stage of development.

In medical field machine learning plays an important role for thyroid disease diagnosis as it has various classification models based on which we can train our model with proper train dataset of the thyroid patient and can predict and give the results in an accurate manner with higher degree of correctness.

Some recent studies from Mumbai have suggested that congenital hypothyroidism is common in India. The disease occurs in 1 part of 2640 new born children, when compared to the worldwide average range of 1 in 3800 considered. Congenital hypothyroidism can lead to serious complications if not detected in early stages. Therefore, the proposed model serves the goal in early detection of thyroid disease.

Based on the obtained test values the health care staff can easily examine the condition of the patient and also skip further clinical examinations if not necessary. Hence, this approach proves to be very much beneficial to the healthcare field. A proper train dataset results into an accurate predicting model therefore reducing the overall cost of the thyroid patient treatment and also saving the time [2]. Classification algorithms are most suitable in decision making and also solving the real-world problems.

**ABOUT THYROID**

The Thyroid is butterfly-shaped endocrine gland which is situated at the base of the human neck. The vital role of the thyroid gland is maintaining and balancing human metabolism and also the growth and development of the human body. The vital tasks performed by thyroid gland are blood circulation, body temperature control, muscle strength and brain functioning. Any damage or improper functioning of the gland may seriously affect the normal human body functioning. Therefore, proper thyroid hormone secretion results into a healthy human body. If there is either low or high secretions of the hormone it will adversely affect the human health.

The Thyroid disorder is the most common endocrine disease across the world. In a survey carried out in India, around 42 million people are suffering from this disease [1]. Thyroid disease is different from other type of endocrine diseases in terms of the mode of treatment relative attainability and the ease of predicting the disease

The high thyroid hormone secretion leads to Hyperthyroidism and low secretion leads to Hypothyroidism. Both the conditions adversely affect the human physiology and the symptoms shown for hyperthyroidism are dry skin, hair thinning, loss of weight, high blood pressure, neck enlargement and short menstrual periods.

The symptoms show for hypothyroidism include the thyroid gland inflammation, weight gain, low blood pressure, heavy menstrual periods and loss of appetite.

These symptoms may get even worse if they are not treated in an early stage. Hence, there is a need for a proper prediction model which helps in diagnosing the patient’s condition in an early stage of the disease.

**EXISTING SYSTEM**

In the data collection stage, small, memory-constrained and low energy-consumption sensors with a short-range communications capability are employed to collect information about the physical environment. Ethernet, WIFI, ZigBee, and wire-based technologies are combined with Transmission Control Protocol/Internet Protocol to connect the objects and users across prolonged distances during data transmission. During the data processing and utilization stage, applications process the data to obtain useful information, and may initiate control commands to act on the physical environment after making decisions based on the collected information. The coordination of diverse technologies, the heterogeneity, and the distributed nature of communications technologies proposed for the IoT by different standards development organizations [4] magnify the threat to end-to-end security in IoT applications.

**DISADVANTAGES OF EXISTING SYSTEM:**

1) Less accuracy

2) Low Efficiency

**PROPOSED SYSTEM:**

The thyroid dataset is taken from Kaggle Machine Learning website. The Database mainly includes the thyroid patient records having all the necessary patient details in it. The patient record has important attributes as mentioned in the Table I. Along with this, the proposed model also takes all the records of the patient’s past clinical history showing in the Fig 1. These include whether the patient is allergic to any particular medicine, whether the patient has undergone any past thyroid surgery and also any recent thyroid test and genetic history of the patient. These also act as the major attributes since they ease the examination of the thyroid patient and reduce the thorough examination by the doctor. This saves time and eases the diagnosis process.

These attributes are stored in a dedicated cloud server which can be made private or hybrid based on the health organization’s need and interest. Among the considered attributes a train dataset is prepared and is given as the input to the classification-based machine learning model. This is a supervised learning method and the designed model will generate the results based on the train dataset values. The proposed model has Decision tree and Naive Bayes algorithm to generate the results. A decision tree is a tree-based algorithm which follows a top-down approach build. ID3 algorithm is used to construct the decision tree. It mainly eliminates any redundant element if present and improves the accuracy of the classification.

The patient’s thyroid stage here is divided into 3 stages i.e., minor, major and critical. The Naïve Bayes algorithm is applied if the Decision Tree returns thyroid true or positive value. The Naïve Bayes algorithm in machine learning is a supervised learning algorithm which is based on Bayes’ Theorem.

It is used for solving classification-based problems. The model can be built fast and it is also cost effective one. In this way our proposed system can make a major contribution in the healthcare field and also generate positive outcomes with good accuracy with a cost and time saving method for the thyroid patients

**ADVANTAGES OF PROPOSED SYSTEM:**

1) High accuracy

2) High efficiency

#### 

#### CHAPTER 2

**SYSTEM STUDY**

##### FEASIBILITY STUDY

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

**Two key considerations involved in the feasibility analysis are**

* **ECONOMICAL FEASIBILITY**
* **TECHNICAL FEASIBILITY**

##### ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

##### TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the

available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for system.

#### CHAPTER 3

##### REQUIREMENT ANALYSIS

The project involved analysing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well-ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

**REQUIREMENT SPECIFICATIONS**

* **Hardware**
* **Software**

**HARDWARE REQUIRMENTS:**

* System    :   i3 or above.
* Ram    :   4 GB.
* Hard Disk : 40 GB

**SOFTWARE REQUIRMENTS:**

* Operating system : Windows8 or Above.
* Coding Language  : python

#### CHAPTER 4

**SYSTEM DESIGN**

##### SYSTEM ARCHITECTURE

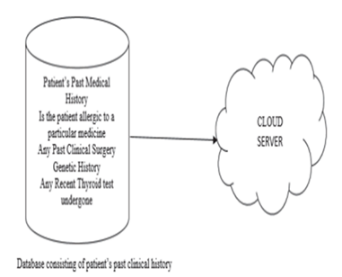


Fig 4.1 System Architecture

##### USE CASE DIAGRAM

##### 

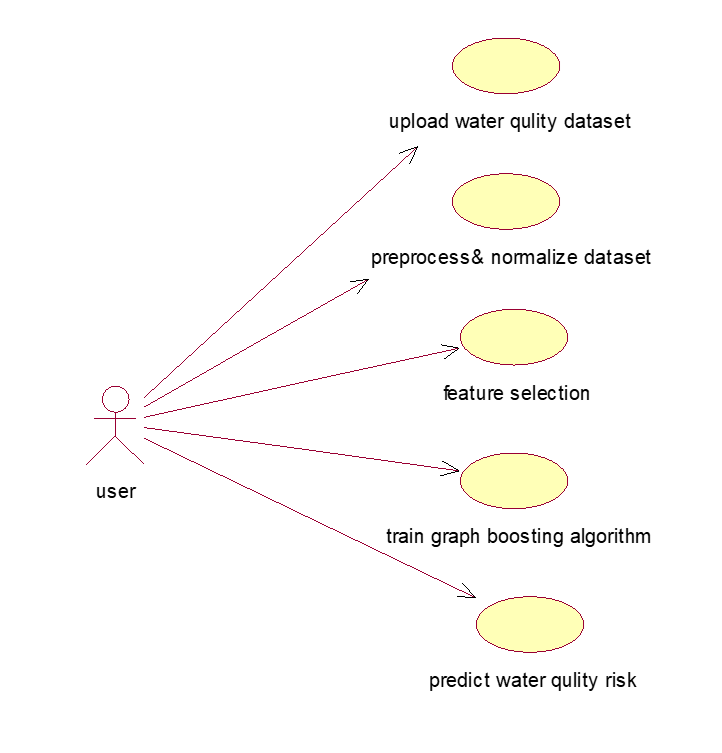


Fig 4.2 Use Case Diagram

##### CLASS DIAGRAM

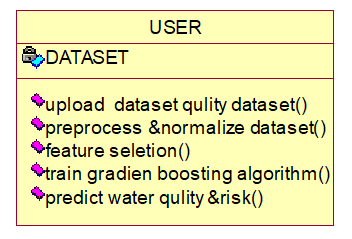
****

Fig 4.3 Class Diagram

**COLLABRATION DIAGRAM:**

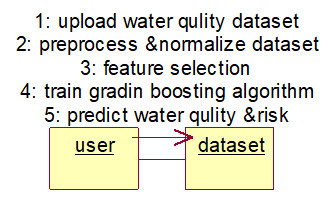


Fig 4.4 Collaboration Diagram

##### SEQUENCE DIAGRAM



Fig 4.5 Sequence Diagram

#### CHAPTER 5

##### MODULES

There are four modules can be divided here for this project they are listed as below

1. Upload Thyroid Dataset
2. Preprocess Dataset
3. Run Naive Bayes Algorithm
4. Run SVM Algorithm
5. Run Random Forest Algorithm
6. Comparison Graph:
7. Predict Disease from Test Data

**MODULE DESCRIPTION:**

1. **Upload Thyroid Dataset**:

using this module, we will upload dataset details and then application will read and display dataset values and then find and plot graph of normal and thyroid patients count

1. **Preprocess Dataset:**

using this module, we will preprocess dataset and then remove missing values and then convert all non-numeric data into numeric data and then shuffle and then split dataset into train and test where application using 80% dataset for training and 20% for testing

1. **Run Naive Bayes Algorithm:**

using this module, we will input 80% dataset to Naïve Bayes to train a model and 20% test data will be applied on trained model to calculate prediction accuracy

1. **Run SVM Algorithm:**

using this module, we will input 80% dataset to SVM to train a model and 20% test data will be applied on trained model to calculate prediction accuracy

1. **Run Random Forest Algorithm**:

using this module, we will input 80% dataset to Random Forest to train a model and 20% test data will be applied on trained model to calculate prediction accuracy

1. **Comparison Graph**:

using this module, we will plot accuracy comparison graph between all algorithms

1. **Predict Disease from Test Data:**

using this module, we will upload test data and then random forest will predict weather test data is normal or contains thyroid disease

#### CHAPTER 6

##### ALGORITHM

**NAIVE BAYES ALGORITHM**

Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. There is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible correlation between the colour, roundness, and diameter features. In many practical applications, parameter estimation for naive

Bayes models uses the method of maximum likelihood; in other words, one can work with the naive Bayes model without accepting Bayesian probability or using any Bayesian methods. Despite their naive design and apparently oversimplified assumptions, naive Bayes classifiers have worked quite well in many complex real-world situations. In 2004, an analysis of the Bayesian classification problem showed that there are sound theoretical reasons for the apparently implausible efficacy of naive Bayes classifiers. Still, a comprehensive comparison with other classification algorithms in 2006 showed that Bayes classification is outperformed by other approaches, such as boosted trees or random forests. An advantage of naive Bayes is that it only requires a small number of training data to estimate the parameters necessary for classification.

#### Support Vector Machine

**“**Support Vector machine” (SVM) is a supervised machine learning algorithm which can be used for both classification and regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n- dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate.

Then, we perform classification by finding the hyper-plane that differentiate the two classes very well (look at the below snapshot). The SVM algorithm is implemented in practice using a kernel. The learning of the hyper plane in a linear SVM is done by transforming the problem using some linear algebra, which is out of the scope of this introduction to SVM.A powerful insight is that the linear SVM can be rephrased using the inner product of any two given observations, rather than the observations themselves. The inner product between two vectors is the sum of the multiplication of each pair of input values. For example, the inner product of the vectors [2, 3] and [5, 6] is 2\*5 + 3\*6 or 28. The equation for making a prediction for a new input using the dot product between the input (x) and each support vector (xi) is calculated as follows: f(x) = B0 + sum (ai \* (x, xi)).

**RANDOM FOREST ALGORITHM**

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

**Source Code**

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

import numpy as np

import os

import pandas as pd

from sklearn.metrics import accuracy\_score

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

from sklearn.metrics import confusion\_matrix

from sklearn.naive\_bayes import GaussianNB

from sklearn import svm

from sklearn.metrics import precision\_score

from sklearn.metrics import recall\_score

from sklearn.metrics import f1\_score

import seaborn as sns

from sklearn.ensemble import RandomForestClassifier

from sklearn.preprocessing import LabelEncoder

from sklearn.preprocessing import StandardScaler

from imblearn.over\_sampling import SMOTE

main = tkinter.Tk()

main.title("Thyroid Disease Detection using Machine Learning Algorithms")

main.geometry("1300x1200")

global filename, rf\_cls, X, Y

global accuracy, precision, recall, fscore

global X\_train, X\_test, y\_train, y\_test

global labels

global label\_encoder, scaler

def upload():

global filename, labels, dataset

filename = filedialog.askopenfilename(initialdir="Dataset")

text.delete('1.0', END)

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

dataset = pd.read\_csv(filename)

text.insert(END,str(dataset)+"\n\n")

text.update\_idletasks()

labels = np.unique("binaryClass")

label = dataset.groupby('binaryClass').size()

label.plot(kind="bar")

plt.xlabel("P (Thyroid Presence) & N (Normal)")

plt.ylabel("Count")

plt.title("Normal & Thyroid Patient Graph")

plt.show()

def preprocess():

text.delete('1.0', END)

global dataset, X, Y, label\_encoder, scaler

global X\_train, X\_test, y\_train, y\_test

label\_encoder = []

dataset.fillna(0, inplace = True)

dataset["age"] = dataset["age"].astype(float)

dataset["TSH"] = dataset["TSH"].astype(float)

dataset["T3"] = dataset["T3"].astype(float)

dataset["TT4"] = dataset["TT4"].astype(float)

dataset["T4U"] = dataset["T4U"].astype(float)

dataset["FTI"] = dataset["FTI"].astype(float)

columns = dataset.columns

types = dataset.dtypes.values

print(dataset.info())

for i in range(len(types)):

name = types[i]

if name == 'object':

le = LabelEncoder()

dataset[columns[i]] = pd.Series(le.fit\_transform(dataset[columns[i]].astype(str)))

label\_encoder.append(le)

#function to calculate all metrics

def calculateMetrics(algorithm, y\_test, predict):

a = accuracy\_score(y\_test,predict)\*100

p = precision\_score(y\_test, predict,average='macro') \* 100

r = recall\_score(y\_test, predict,average='macro') \* 100

f = f1\_score(y\_test, predict,average='macro') \* 100

accuracy.append(a)

precision.append(p)

recall.append(r)

fscore.append(f)

text.insert(END,algorithm+" Accuracy : "+str(a)+"\n")

text.insert(END,algorithm+" Precision : "+str(p)+"\n")

text.insert(END,algorithm+" Recall : "+str(r)+"\n")

text.insert(END,algorithm+" FScore : "+str(f)+"\n\n")

conf\_matrix = confusion\_matrix(y\_test, predict)

label = ["Normal", "Throid Detected"]

ax = sns.heatmap(conf\_matrix, xticklabels = label, yticklabels = label, annot = True, cmap="viridis" ,fmt ="g");

ax.set\_ylim([0,len(label)])

plt.title(algorithm+" Confusion matrix")

plt.ylabel('True class')

plt.xlabel('Predicted class')

plt.show()

def runNaiveBayes():

text.delete('1.0', END)

global X\_train, X\_test, y\_train, y\_test

global accuracy, precision, recall, fscore

accuracy = []

precision = []

recall = []

fscore = []

nb\_cls = GaussianNB()

nb\_cls.fit(X\_train, y\_train)

predict = nb\_cls.predict(X\_test)

calculateMetrics("Naive Bayes", y\_test, predict)

def runSVM():

svm\_cls = svm.SVC()

svm\_cls.fit(X\_train, y\_train)

predict = svm\_cls.predict(X\_test)

calculateMetrics("SVM", y\_test, predict)

def runRF():

global rf\_cls

rf\_cls = RandomForestClassifier()

rf\_cls.fit(X\_train, y\_train)

predict = rf\_cls.predict(X\_test)

calculateMetrics("Random Forest", y\_test, predict)

def graph():

df = pd.DataFrame([['Naive Bayes','Precision',precision[0]],['Naive Bayes','Recall',recall[0]],['Naive Bayes','F1 Score',fscore[0]],['Naive Bayes','Accuracy',accuracy[0]],

['SVM','Precision',precision[1]],['SVM','Recall',recall[1]],['SVM','F1 Score',fscore[1]],['SVM','Accuracy',accuracy[1]],

['Random Forest','Precision',precision[2]],['Random Forest','Recall',recall[2]],['Random Forest','F1 Score',fscore[2]],['Random Forest','Accuracy',accuracy[2 ],columns=['Parameters','Algorithms','Value'])

df.pivot("Parameters", "Algorithms", "Value").plot(kind='bar')

plt.show()

def predict():

text.delete('1.0', END)

global rf\_cls, scaler, label\_encoder

filename = filedialog.askopenfilename(initialdir="Dataset")

predict = rf\_cls.predict(testData)

print(predict)

for i in range(len(predict)):

if predict[i] != 1:

print(str(i)+" "+str(predict[i]))

for i in range(len(predict)):

pred = predict[i]

if pred == 0:

text.insert(END,"Test Data = "+str(dataset[i])+"=====> Predicted AS NORMAL\n\n")

if pred == 0:

text.insert(END,"Test Data = "+str(dataset[i])+"=====> Predicted AS THYROID DETECTED\n\n")

font = ('times', 16, 'bold')

title = Label(main, text='Thyroid Disease Detection using Machine Learning Algorithms')

title.config(bg='firebrick4', fg='dodger blue')

title.config(font=font)

title.config(height=3, width=120)

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

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

text=Text(main,height=24,width=150)

scroll=Scrollbar(text)

text.configure(yscrollcommand=scroll.set)

text.place(x=40,y=120)

text.config(font=font1)

font1 = ('times', 13, 'bold')

uploadButton = Button(main, text="Upload Thyroid Dataset", command=upload, bg='#ffb3fe')

uploadButton.place(x=50,y=600)

uploadButton.config(font=font1)

processButton = Button(main, text="Preprocess Dataset", command=preprocess, bg='#ffb3fe')

processButton.place(x=370,y=600)

processButton.config(font=font1)

nbButton1 = Button(main, text="Run Naive Bayes Algorithm", command=runNaiveBayes, bg='#ffb3fe')

nbButton1.place(x=610,y=600)

nbButton1.config(font=font1)

svmButton = Button(main, text="Run SVM Algorithm", command=runSVM, bg='#ffb3fe')

svmButton.place(x=900,y=600)

svmButton.config(font=font1)

rfButton = Button(main, text="Run Random Forest Algorithm", command=runRF, bg='#ffb3fe')

rfButton.place(x=50,y=650)

rfButton.config(font=font1)

graphButton = Button(main, text="Comparison Graph", command=graph, bg='#ffb3fe')

graphButton.place(x=360,y=650)

graphButton.config(font=font1)

predictButton = Button(main, text="Predict Disease from Test Data", command=predict, bg='#ffb3fe')

predictButton.place(x=610,y=650)

predictButton.config(font=font1)

main.config(bg='LightSalmon3')

main.mainloop()

#### CHAPTER 7

##### SYSTEM TEST

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

**TYPES OF TESTS**

###### Unit testing

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

program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration.

###### Integration testing

###### 

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**Functional Test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centred on the following items:

Valid Input:

Identified classes of valid input must be accepted. Invalid Input: identified classes of invalid input must be rejected. Functions: identified functions must be exercised.

Output:

Identified classes of application outputs must be exercised. Systems/Procedures:

interfacing systems or procedures must be invoked.

###### System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

###### 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 purpose. It is used to test areas that cannot be reached from a black box level.

###### Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in

which the software under test is treated, as a black box. You cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

###### Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase

of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

###### Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

#### CHAPTER 8

##### INPUT AND OUTPUT DESIGN

###### INPUT DESIGN

The input design is the link between the information system and the user. It

comprises the developing specification and procedures for data preparation and thosesteps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides

security and ease of use with retaining the privacy. Input Design considered the following things:

* What data should be given as input?
* How the data should be arranged or coded?
* The dialog to guide the operating personnel in providing input.
* Methods for preparing input validations and steps to follow when error occur.

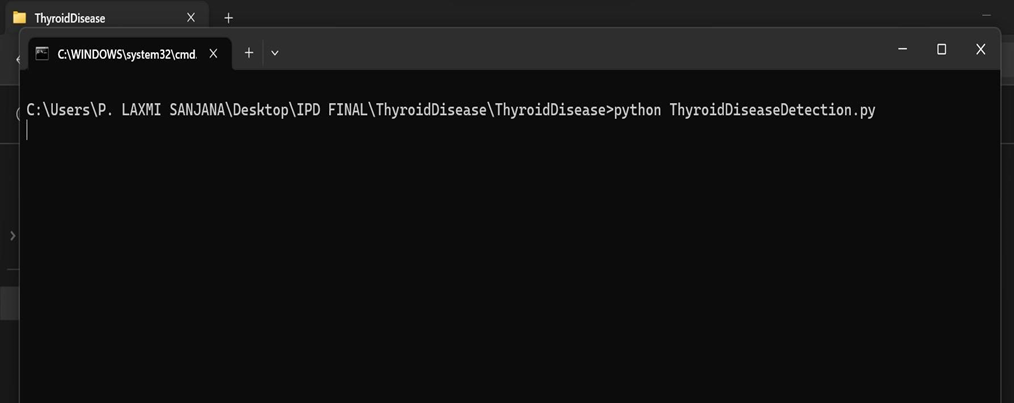
###### OUTPUT DESIGN

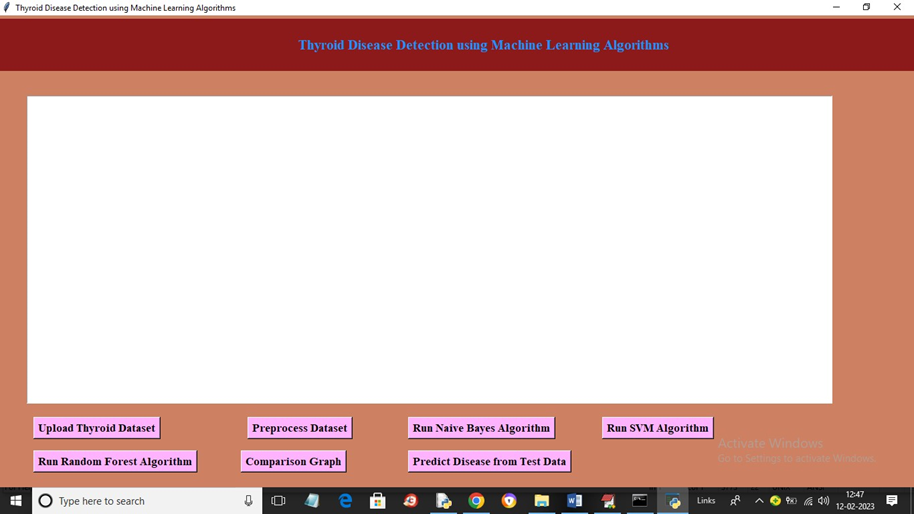
###### 

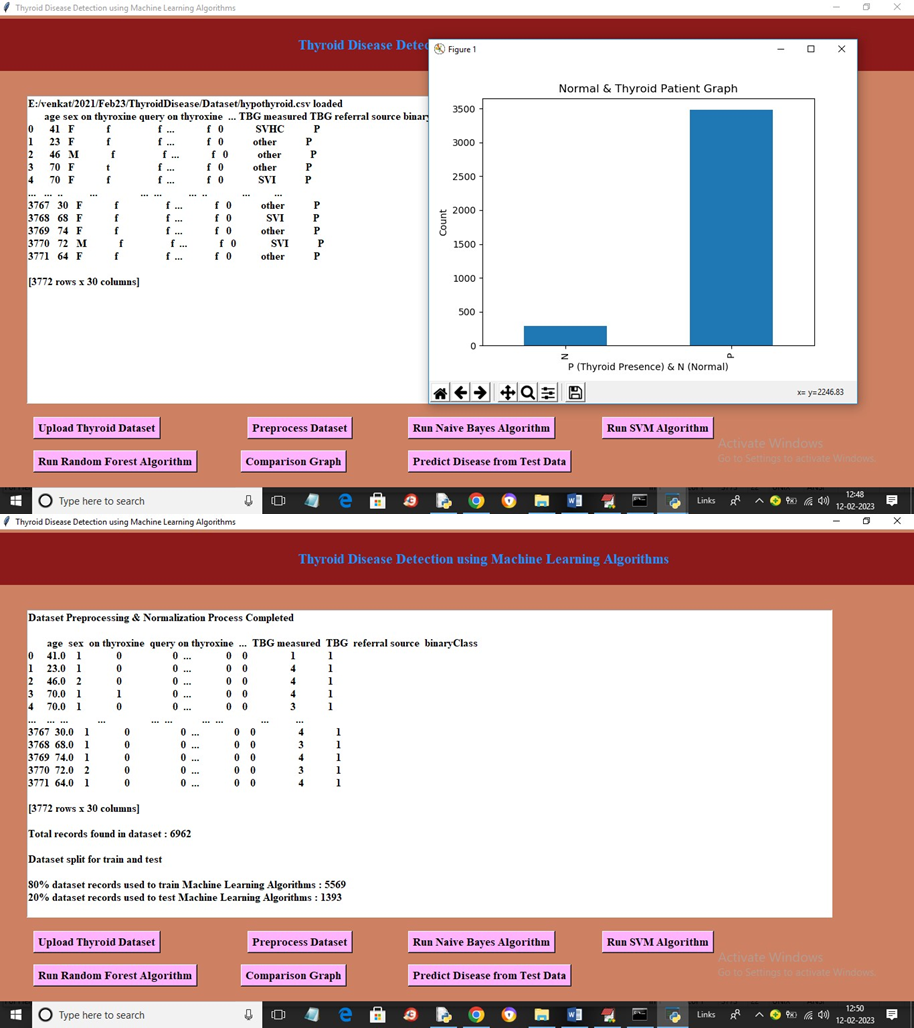
A quality output is one, which meets the requirements of the end user and

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

#### OUTPUT SCREENS

****

****

****

#### 

#### 

#### 

#### 

#### 

#### 

#### CHAPTER 9

##### CONCLUSION

Thus, the proposed work will be very much useful to identify the thyroid disease in a patient at an early stage using classification-based machine learning techniques. These algorithms give various levels of precision and accuracy. These methods also aid in decreasing the unwanted redundant data from the patient’s database. The algorithms used in the proposed model are cost effective and also have good output performance and speed. These classification methods make the treatment of the thyroid patient simple by reducing further complex procedures with an affordable price.

**CHAPTER 10**

#### REFERENCES

1. <https://www.kaggle.com/kumar012/hypothyroid>
2. Bibi Amina Begum and Dr. Parkavi “Prediction of thyroid Disease Using Data Mining Techniques” 5Th International Conference on Advanced Computing & Communication Systems (ICACCS), 2019
3. Ankith Tyagi, Ritika Mehra, Aditya Saxena “Interactive Thyroid Disease Prediction System Using Machine Learning Technique” 5th IEEE International Conference on Parallel, Distributed and Grid Computing (PDGC-2018), 20-22 Dec, 2018, Solan, India
4. Aswathi A K and Anil Antony “An Intelligent System for Thyroid Disease Classification and Diagnosis” Proceedings of the 2nd International Conference on Inventive Communication and Computational Technologies (ICICCT 2018) IEEE Xplore Compliant - Part Number: CFP18BAC-ART; ISBN:978-1-5386-1974-2