ANTICIPATING POTENTIAL ILLNESS THROUGH PREDICTIVE MODELS

A Minor Project Report

in partial fulfillment of the degree

Bachelor of Technology
in
Computer Science & Artificial Intelligence

By

(2103A52005)	Basani Shashanka
(2103A52027)	Polsani Chaithra Sri
(2103A52028)	Chidurala Ashwini
(2103A52085)	Katha Sravani
(2103A52098)	Munja Sriharsha

Under the Guidance of

Dr Pramod Kumar Poladi

Submitted to

SCHOOL OF COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE
SR UNIVERSITY, ANANTHASAGAR, WARANGAL

April, 2024.

SCHOOL OF COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE

CERTIFICATE

This is to certify that this project entitled **"Anticipating potential illness through predictive models "** is the bonafied work carried out by **Polsani Chaithra Sri, Chidurala Ashwini , Basani Shashanka, Katha Sravani , Munja Sriharsha** as a Minor Project for the partial fulfillment to award the degree **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE** during the academic year 2023-2024 under our guidance and Supervision.

Dr Pramod Kumar Poladi
Assistant Prof,
SR University,
Ananthasagar, Warangal.

Dr. M.Sheshikala
Assoc. Prof. & HOD (CSE),
SR University,
Ananthasagar, Warangal.

External Examiner

ACKNOWLEDGEMEN

We owe an enormous debt of gratitude to our project guide **Dr. Dr Pramod Kumar Poladi, Assistant Professor** as well as Head of the CSE Department **Dr. M.Sheshikala, Associate Professor** for guiding us from the beginning through the end of the Minor Project with their intellectual advices and insightful suggestions. We truly value their consistent feedback on our progress, which was always constructive and encouraging and ultimately drove us to the right direction.

We express our thanks to the project co-ordinators **Dr. P Praveen, Assoc. Prof** for their encouragement and support.

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved Dean, **Dr.Indrajeet Gupta,** for his continuous support and guidance to complete this project in the institute.

Finally, we express our thanks to all the teaching and non-teaching staff of the department for their suggestions and timely support.

Polsani Chaithra Sri

Chidurala Ashwini

Basani Shashanka

Munja Sri Harsha

Katha Sravani

ABSTRACT

This study aims to develop a machine learning model for predicting 42 diseases based on 132 symptoms. The goal is to train a model on the training data and evaluate its performance on the testing data. However, existing methods of disease prediction are often limited by the availability and quality of health data, as well as the complexity and diversity of diseases. In this paper, we propose a novel machine learning framework for disease prediction and improvement, based on the integration of data sources, such as symptoms. This study presents the development of a machine learning model for disease prediction across a spectrum of medical conditions. the model aims to accurately forecast the likelihood of individuals developing various diseases, including but not limited to cardiovascular diseases, diabetes, cancer, and neurological disorders. The proposed model integrates state-of-the-art machine learning algorithms, including Random forest, voting model, ANN to effectively capture complex patterns and interactions within the data. By accurately predicting diseases based on symptoms, this study can significantly assist physicians and improve healthcare outcomes. We demonstrate that our framework can achieve high accuracy and robustness in disease prediction, as well as provide meaningful and actionable insights for disease prevention and management.

TABLE OF CONTENTS

INTRODUCTION

Numerous industries, including medical science, have advanced significantly as a result of the increased interest in machine learning and deep learning. Using cutting-edge technology and data analytics, disease prediction has become a critical area in healthcare that is changing the way we think about early detection and prevention. Healthcare practitioners may now predict an individual's risk of having a certain disease by using genetic profiling, big data analytics, and machine learning algorithms. This allows for preemptive steps to be taken to reduce risks and improve public health outcomes. At the heart of contemporary healthcare is disease prediction, which has revolutionary possibilities to improve population health outcomes, customize medical interventions, and strengthen preventative care. The field of disease prediction is set to grow as medical science and technology continue to progress and our understanding of disease mechanisms develops.

There is an overwhelming amount of data coming from many sources into the healthcare sector every day. Large-scale data collection and storage by internet behemoths like Google and Facebook results in big data. With the utilization of this data, important insights that could enhance healthcare delivery and bring in large sums of money can be produced.

Technically sophisticated software and apps that can quickly and affordably analyze large amounts of data using high-end processing power are needed to handle big data. Making sense of this massive amount of data can be aided by the application of innovative fusion algorithms and artificial intelligence (AI) algorithms. It can be a tremendous accomplishment to automate decision-making through the use of machine learning (ML) techniques like neural networks and other AI technologies.

1.1 EXISTING SYSTEM

Several models currently in use make use of clinical data or patient-reported symptoms to forecast different diseases using symptom-based algorithms. These models do, however, frequently have shortcomings, such as poor accuracy brought on by symptom overlap between many illnesses, which can result in incorrect diagnoses or false alarms. Furthermore, individual differences in symptom presentation or illness development may not be taken into account by symptom-based models, which lessens their usefulness in individualized healthcare settings. Moreover, biases and errors can be introduced into the prediction process by depending on self-reported symptoms or insufficient clinical data, which compromises the validity of these models in practical applications.

Restricted Scope: A large number of current models concentrate on forecasting a particular illness or a small subset of related diseases. This methodology could fail to consider the intricacy of shared symptoms among different illnesses, resulting in partial or imprecise forecasts.

Symptom Overlap: Predictive models may be confused by the common symptoms that many diseases share. The inability of current methods to reliably distinguish between illnesses with comparable symptom patterns raises the possibility of a misdiagnosis or a delay in receiving treatment.

Diversity and Quality of Data: The calibre and variety of the training data greatly influences the efficacy of symptom-based prediction models. Predictions' generalizability and reliability may be impacted by biases, inadequate records, or a lack of representation of particular demographics or geographic areas in the datasets that are currently available.

1.2 PROPOSED SYSTEM

The dataset we have considered consists of 132 symptoms, the combination or permutations of which leads to 41 diseases. Based on the 4920 records of patients, we aim to develop a prediction model that takes in the symptoms from the user and predicts the disease he is more likely to have. If symptom is present indicated by 1 if not present indicated by 0.Our proposed sysytem mainly focuses to develop a machine learning model for the below cases.

- Creating a disease prediction algorithm that can recognize different health issues based on different symptom profiles.
- Developing a flexible tool that provides predictions for multiple scenarios, straying from the current models' emphasis on a particular illness.
- To improve accuracy and early detection skills, make use of extensive datasets and cutting edge machine learning approaches.
- Enhance patient outcomes by allowing early detection to provide prompt intervention for a variety of disorders.
- Facilitate clinical diagnostic workflows by giving medical staff a dependable and effective tool for making decisions.

2. LITERATURE SURVEY

SNO	Authors	Title	Demerits	Future Scope
1	Sneha Grampurohit , Chetan sagarnal.	DiseasePrediction usingMachine Learning Algorithms.	This may require additional resources and expertise to ensure the model remains effective and up-to-date.	Our future work investigates that how the proposed method can be extended to be applicable to the other types of datasets in medical domain.
2	Dhiraj Dahiwade, ,Prof. Gajanan Patle,Prof. Ektaa Meshram	Designing Disease Prediction Model Using Machine Learning Approach.	Here small volume data used for prediction like symptoms or previous knowledge obtained from the physical diagnosis.	They could not consider large dataset, now a days medical data is growing so needs to classify that and classification of that data is challenging.
3	Shahadat Uddin1 ,Arif Khan1,Md EkramulHossain1 and Mohammad Ali Moni.	Comparing different supervised machine learning algorithms for disease prediction.	The limitation is that they did not consider any sub classifications or variants of any of the algorithms ,considered in this study.	There is a need for more comprehensive comparative studies that evaluate the performance of different supervised machine learning algorithms for disease prediction.
4	Mehrbakhsh Nilashi, Othman bin Ibrahim, Hossein Ahmadi, Leila Shahmoradi.	An Analytical Method for Diseases Prediction Using Machine Learning Technique.	All of the approaches used in this study, may not be applicable to other diseases classification problems	More attention should be paid to the dataset for disease classification and prediction using the incremental machine learning approaches.

5	K.Arumugama ,MohdNavedb, PriyankaP. Shindec,Orlan doLeivaChauc ad,Antonio HuamanOsorio e,Tatiana GonzalesYanac	Multiple disease prediction using Machine learning algorithms	Limited availability of high-quality labeled dataset, especially for rare diseases.	Research on interpretable machine learning models that provide transparent explanations for disease predictions.
6	Zengchen Yu1, KeWang,Zhibo Wan1, Shuxuan Xie1, Zhihan Lv.	Popular deep learning algorithms for disease prediction.	Poor interpretability,Data imbalance,Data quality issues,Too little data.	Incorporating Digital Twins,Promoting precision medicine.
7	Abrar Yaqoob 1, Rabia Musheer Aziz 1 , Navneet Kumar Verma 1 and Pavan Kumar.	A Review on Nature-Inspired Algorithms for Cancer Disease Prediction and Classification.	Time-consuming and challenging.	Advancing towards personalized cancer prediction models that consider individual patient characteristics, including genetic profiles, environmental factors, and treatment history.
8	Weicheng Sun1, Ping Zhang1,Zilin Wang 1 and Dongxu Li.	Prediction of Cardiovascular Diseases based on Machine Learning.	ML models may overfit to the training data, capturing noise rather than true underlying patterns.	ML models can be trained to tailor predictions and treatment recommendationsto individual patients based on their unique risk factors and characteristics.
9	Varsha Nemade1,Vish al Fegade	Machine Learning Techniques for Breast Cancer Prediction	Machine learning models trained on specific datasets may not generalize well to other populations which can	The future scope of this work is to perform hyper parameter tuning to improve the performance of the model.

			limit their applicability and effectiveness in real-world clinical settings.	
10	Gyanendra Chaubey1, Dhananjay Bisen1, Siddharth Arjaria1, Vibhash Yadav1	Thyroid Disease Prediction Using Machine Learning Approache	Data quality and availability.Limited scope.	Conducting more rigorous validation and testing of machine learning models for thyroid disease prediction,

3. DESIGN

3.1 REQUIREMENT SPECIFICATION

Software requirements

Programming Language: Because of its large machine learning library and user-friendliness,we are using Python for project code development.

Development Environment: The main environment for development will be Google Colab. It provides free access to TPUs and GPUs, which can greatly accelerate machine learning activities including model training.

Machine Learning Libraries: To create and train machine learning models, such as Artificial Neural Networks (ANN) for illness prediction, use libraries such as scikit-learn, TensorFlow, and Keras.

Version Control: To facilitate smooth communication, track changes, and preserve the integrity of the codebase, GitHub will be utilized for version control.

Documentation: Detailed documentation, such as code explanations, API references, and installation instructions, can be created using word document and power point .

Hardware requirements

CPU: For effective computing, a laptop with a contemporary multicoreprocessor. For the best performance, look for AMD Ryzen series or Intel Core i5 or i7 processors.

RAM: To properly manage data processing and model training, using 8GB to 16GB of RAM. When working with larger datasets and complicated models, more RAM could be useful.

Storage: For faster data access and improved overall performance, ideally choosed a laptop with solid-state drive (SSD) storage. It is advised to have 256GB or more of storage space in order to hold project files and datasets.

GPU: Having a dedicated GPU on the laptop can be helpful for local development and testing even though Google Colab will enable access to GPUs for model training. GPUs from NVIDIA, like the GeForce

3.2 UML DIAGRAMS

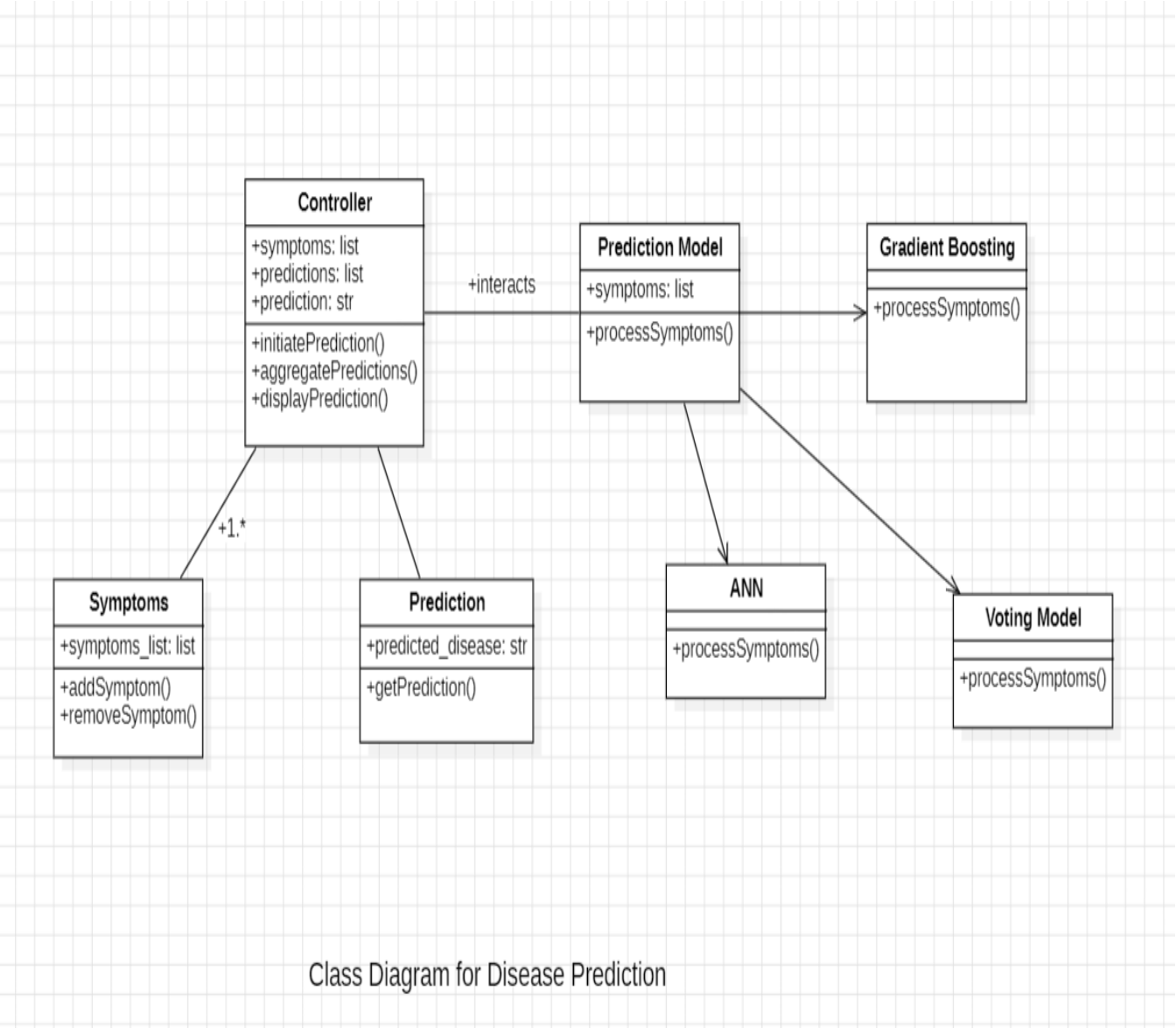

FIG:3.2.1CLASS DIAGRAM

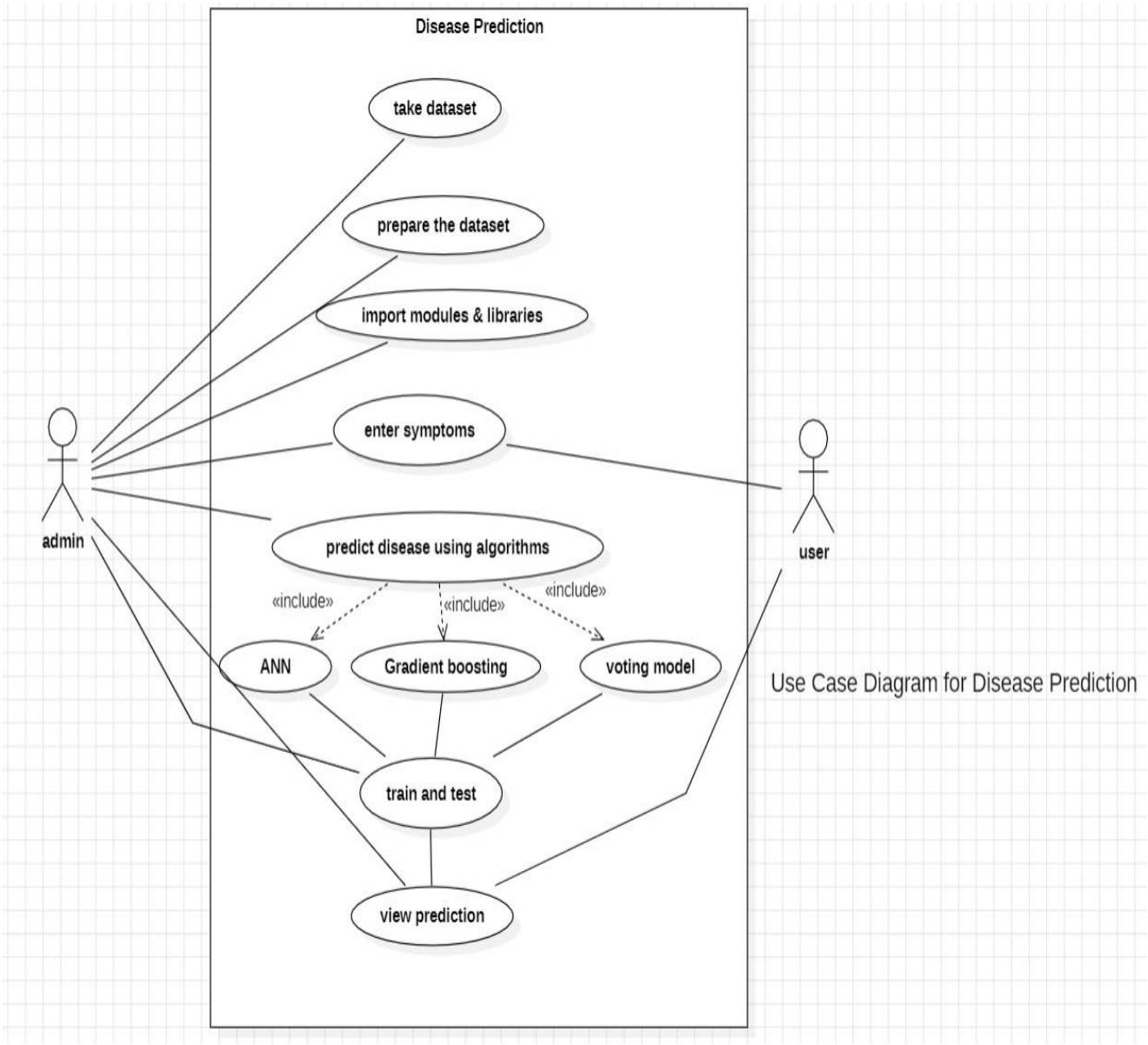

FIG 3.2.2 Use case diagram

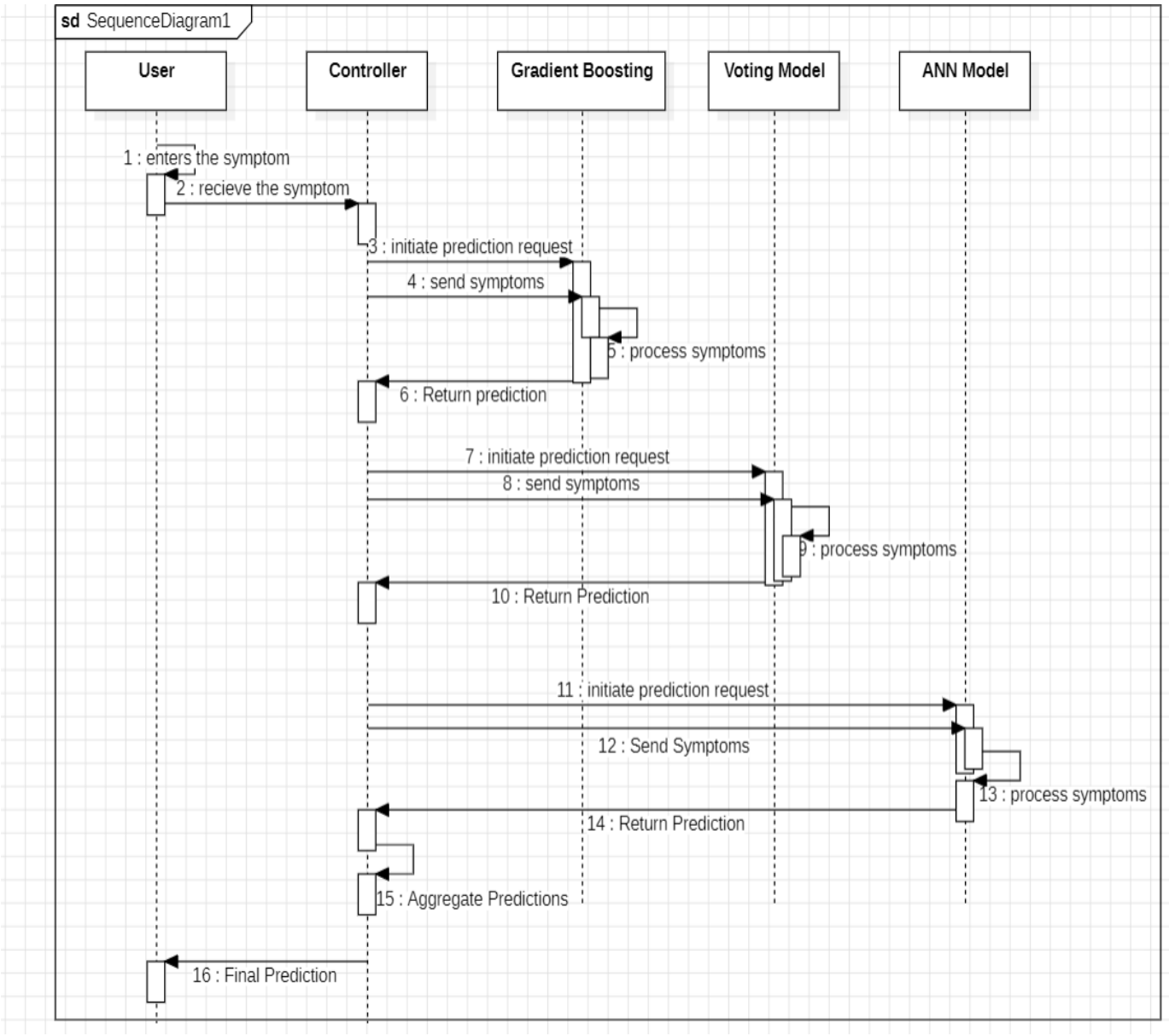

FIG:3.2.3 sequence diagram

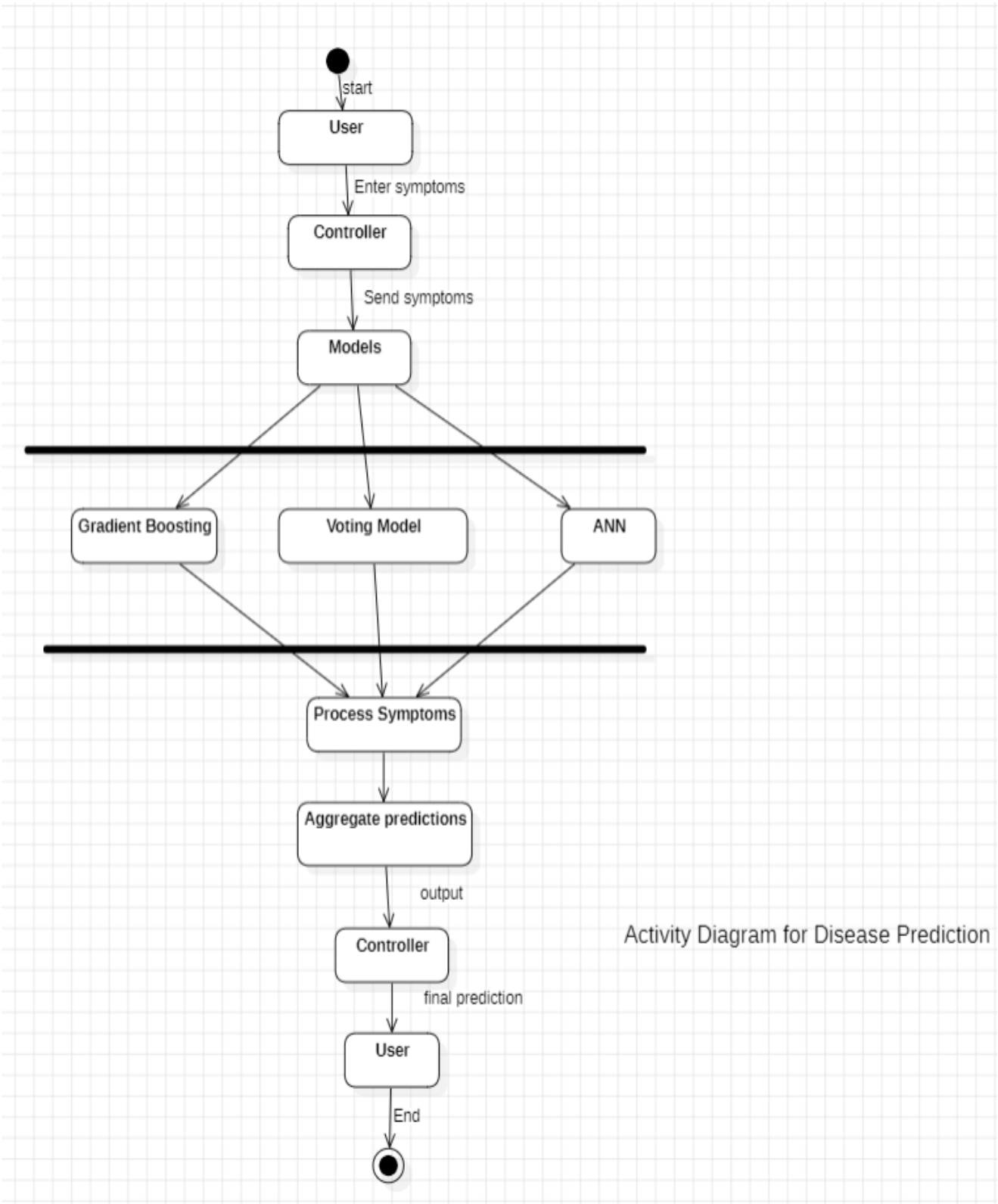

Fig:3.2.4:activity diagram

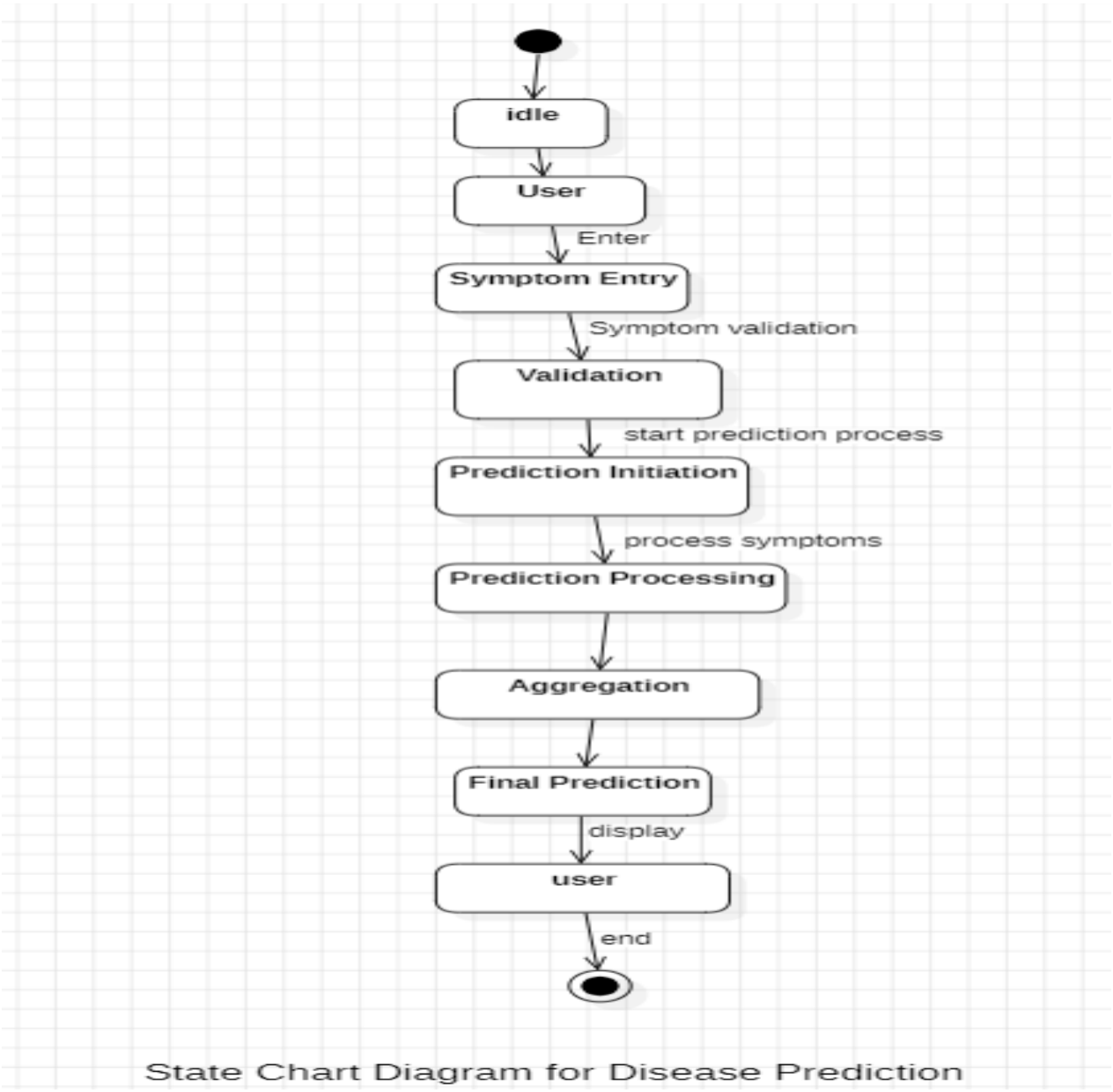

FIG:3.2.5 State chart diagram

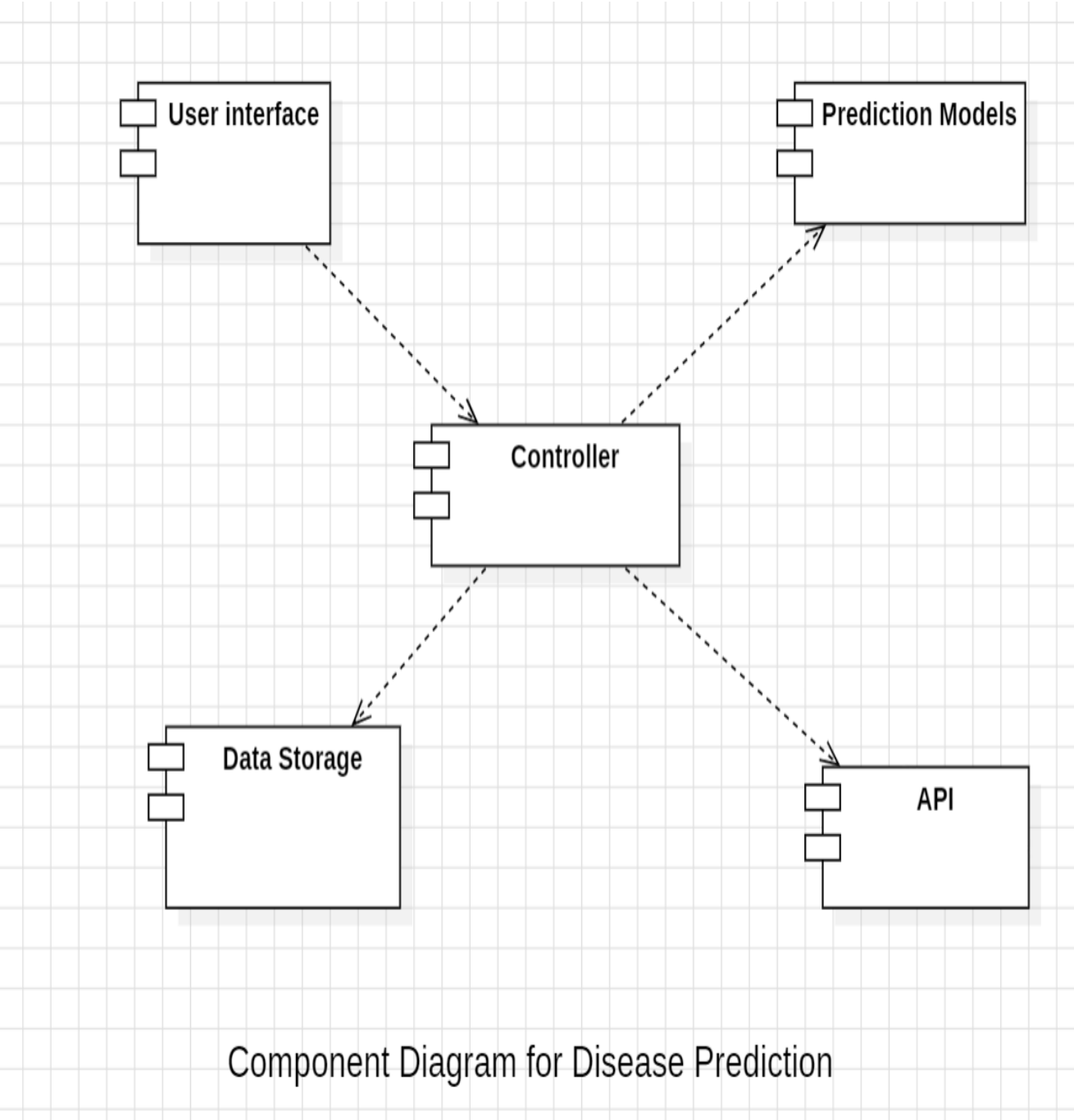

FIG:3.2.6 Component diagram

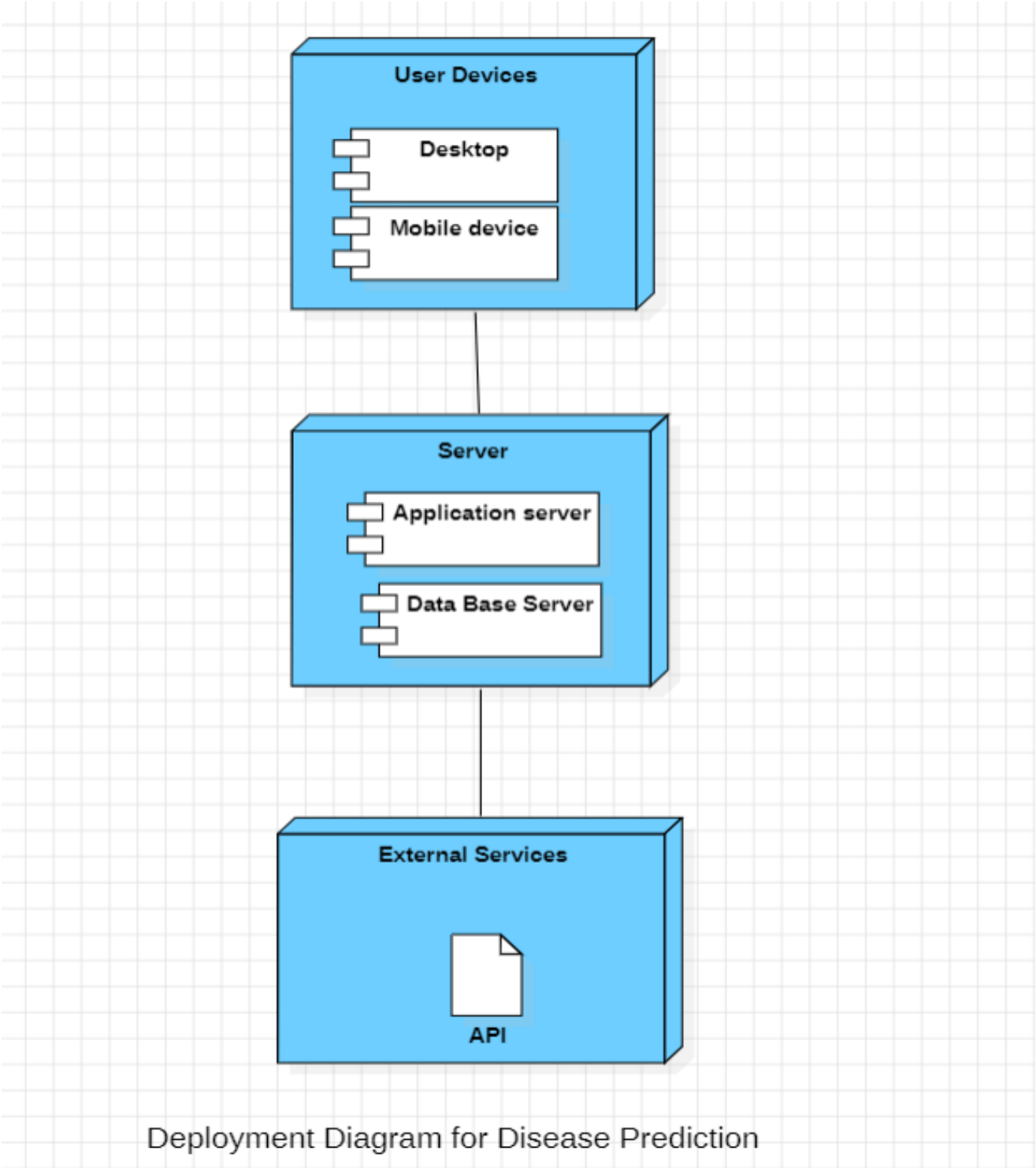

FIG:3.2.7 Deployment diagram

4. IMPLEMENTATION

4.1 MODEL ARCHITECTURE

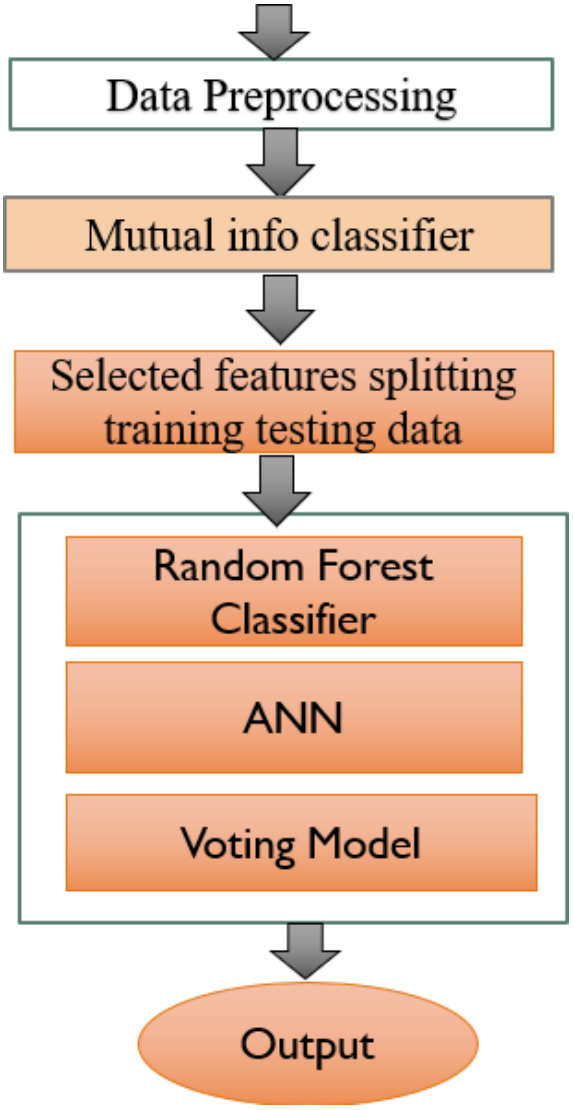

Fig:4.1.1 Model architecture

4.2 MODEL IMPLEMENTATION

Input (Symptoms): Gathering a dataset containing records of patients, their symptoms, and corresponding diagnosed diseases.Preprocess the data by handling missing values, encoding categorical variables, and normalizing numerical features.While designing the model we have assumed that the user has a clear idea about the symptoms he is experiencing. The Prediction developed considers 50 symptoms amidst which the user can give the symptoms his processing as the input.

Data preprocessing : The data mining technique that transforms the raw data or encodes the data to a form which can be easily interpreted by the algorithm is called data preprocessing.

The preprocessing techniques used in the presented work are: Data Cleaning: Data is cleansed through processes such as filling in missing value, thus resolving the inconsistencies in the data.

Data Reduction: The analysis becomes hard when dealing with huge database. Hence, we eliminate those independent variables(symptoms) which might have less or no impact on the target variable(disease). In the present work, 50 of 132 symptoms closely related to the diseases are selected.

Feature Engineering: Compute the Mutual Information scores between each symptom and the target variable (disease).Selecting the top features based on Mutual Information scores (e.g., top 50, out of 132 features).Subset the dataset to include only the selected features.

Data Splitting

Split the dataset into training and testing sets (e.g., 80% training, 20% testing) to evaluate the model's performance.

Models selected : The system is trained to predict the diseases using three algorithms Voting Classifier Random forest Classifier Naïve Bayes Classifier ANN .A comparative study is presented at the end of work, thus analyzing the performance of each algorithm of the considered database.

Model Building

Mutual Information Classifier:

Train a Mutual Information classifier using the selected features.Tune the hyperparameters of the Mutual Information classifier using cross-validation.

Artificial Neural Network (ANN):

Design an ANN architecture suitable for the classification task with an input layer, hidden layers, and an output layer.Train the ANN model using the training data.

Random Forest:

Instantiate a Random Forest classifier with appropriate hyperparameters.

Train the Random Forest model using the training data.

Model Evaluation

Evaluate the performance of each model using metrics such as accuracy, precision, recall, and F1-score on the test dataset.Compare the performance of the models to identify the best-performing one.

Voting Model

Combine the predictions of the Mutual Information classifier, ANN, and Random Forest using a Voting Classifier.Evaluate the performance of the Voting model using the same metrics as before.

Output(diseases): Once the system is trained with the training set using the mentioned algorithms a rule set is formed and when the user the symptoms are given as an input to the model, those symptoms are processed according the rule set developed, thus making classifications and predicting the most likely disease.

5. TESTING

5.1 TEST CASES

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from IPython.display import display
from sklearn.preprocessing import LabelEncoder
from sklearn.ensemble import RandomForestClassifier

from sklearn.model_selection import train_test_split
from sklearn.feature_selection import SelectKBest, chi2
from sklearn.naive_bayes import BernoulliNB
from sklearn.metrics import accuracy_score, classification_report
import warnings

%matplotlib inline
 df= pd.read_csv('/content/Training.csv')
import pandas as pd

# Assuming dftrain is your DataFrame
# Reset the index to synchronize it with the actual columns
df.reset_index(drop=True, inplace=True)

# Check for missing values after resetting index
missing_values = df.isnull().sum()

# Print the missing values
print(missing_values)
df.isnull().sum()
```

```
df.isnull().sum()
import pandas as pd

# Assuming dftrain is your DataFrame
# Replace 'dftrain' with your actual DataFrame name

# Check for columns without names
empty_columns = [col for col in df.columns if not col.strip()]

if empty_columns:
    print("Columns without names:")
    for col in empty_columns:
      print(col)
else:
  print("All columns have names in the DataFrame.")
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay

# Example true labels and predicted labels
true_labels = np.array([0, 1, 0, 1, 1, 0, 1, 0, 1, 1])
predicted_labels = np.array([0, 1, 0, 1, 0, 1, 1, 0, 1, 1])

# Compute confusion matrix
cm = confusion_matrix(true_labels, predicted_labels)

# Display confusion matrix
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=[0, 1])
disp.plot(cmap=plt.cm.Blues)
plt.title('Confusion Matrix')
plt.xlabel('Predicted Label')
plt.ylabel('True Label')
```

```
plt.show()

import seaborn as sns

# Assuming dftrain is your DataFrame
columns = list(df.columns)

# Set a Seaborn style
sns.set(style="whitegrid")

# Define colors
…                   f'{height/total:.1%}', ha="center", fontsize=10)

# Adjust layout
plt.tight_layout()
plt.show()
sorted(df.prognosis.unique())
df[df.duplicated(subset = None, keep = False)]
```

Exploratory Data Analysis

```
from collections import Counter
count = Counter(df['prognosis'])
count.items()
# Assuming 'prognosis' is a categorical column, convert it to categorical data type
df['prognosis'] = df['prognosis'].astype('category')

# count of each target class label
plt.figure(figsize=(30, 5))
ax = sns.countplot(data=df, x='prognosis', palette='PuBu')
ax.set_xticklabels(ax.get_xticklabels(), rotation=40, ha="right")
plt.show()
# Assuming 'columns' is a list of column names in your dataframe
```

```
colors = ['red', 'green']

# Set Seaborn style
sns.set(style="whitegrid")

…
    # Show plot
    plt.show()
# Assuming `columns` is a list of symptom names

colors = ['#fffb08', '#fb08ff']  # Red and Green colors
# barplot of the count for all symptoms' absence and presence
for i in columns:
    fig, ax = plt.subplots(figsize=(8, 6))  # Adjust figsize for better visibility
    bar = df.groupby(i).size().plot(kind='bar', color=colors, ax=ax)

…    for p in bar.patches:
        plt.annotate(str(p.get_height()), (p.get_x() + p.get_width() / 2., p.get_height()),
                     ha='center', va='center', xytext=(0, 10), textcoords='offset points', fontsize=10)

    plt.show()
# Apply label encoding to the 'prognosis' column
label_encoder = LabelEncoder()
df['prognosis'] = label_encoder.fit_transform(df['prognosis'])

print(df)
from sklearn.model_selection import train_test_split
from sklearn.feature_selection import SelectKBest, mutual_info_classif
from sklearn.ensemble import RandomForestClassifier, VotingClassifier
from sklearn.naive_bayes import BernoulliNB
from sklearn.neural_network import MLPClassifier
```

```
from sklearn.metrics import accuracy_score, classification_report
import warnings

# Optional: Ignore warnings to avoid excessive output
…print(classification_report(y_test, ann_pred))

print("\nVoting Classifier Report:")
print(classification_report(y_test, voting_pred))



    # Define models and model names for demonstration purposes (you should use your own trained models)
models = [rf_classifier, nb_classifier, ann_classifier]
model_names = ["Random Forest", "Naive Bayes", "ANN"]

# Initialize lists to store metric values
accuracies = []
precisions = []
recalls = []
f1_scores = []
for model in models:
  # Make predictions on test data
  y_pred = model.predict(X_test_selected)

  # Calculate accuracy, precision, recall, and F1-score
…ax.set_xticklabels(model_names)
ax.set_ylabel('Metric Value')
ax.set_title('Comparison of Model Performance Metrics')
ax.legend()

plt.show()
# Calculate the midpoint to split the features list into two equal parts
```

```
midpoint = len(features) // 2

# Split the features list into two groups
group1_features = features[:midpoint]
group2_features = features[midpoint:]

…# Plot the correlation matrix for group 2
plt.figure(figsize=(12, 10))
sns.heatmap(corr_matrix2, cmap='coolwarm', annot=True, fmt=".2f", linewidths=0.5)
plt.title(f"Correlation Matrix for Group 2 ({len(group2_features)} Features)")
plt.show()
# Calculate symptom prevalence
symptom_prevalence = df.iloc[:, :-1].mean()

# Plot bar chart
plt.figure(figsize=(10, 8))
symptom_prevalence.plot(kind='bar')
plt.title("Symptom Prevalence")
plt.xlabel("Symptom")
plt.ylabel("Prevalence")
plt.show()
# Plot grouped bar charts of symptoms prevalence per disease
symptoms_grouped = df.groupby('prognosis').mean()

# Plot for each disease
diseases = symptoms_grouped.index
num_diseases = len(diseases)

# Plot each symptom's prevalence in each disease
fig, axes = plt.subplots(num_diseases, 1, figsize=(12, 4 * num_diseases))
for i, disease in enumerate(diseases):
```

```
  symptoms_grouped.loc[disease].plot(kind='bar', ax=axes[i])
  axes[i].set_title(f"Symptom Prevalence in {disease}")
  axes[i].set_xlabel("Symptom")
  axes[i].set_ylabel("Prevalence")

plt.tight_layout()
plt.show()
```

5.2 TEST RESULTS

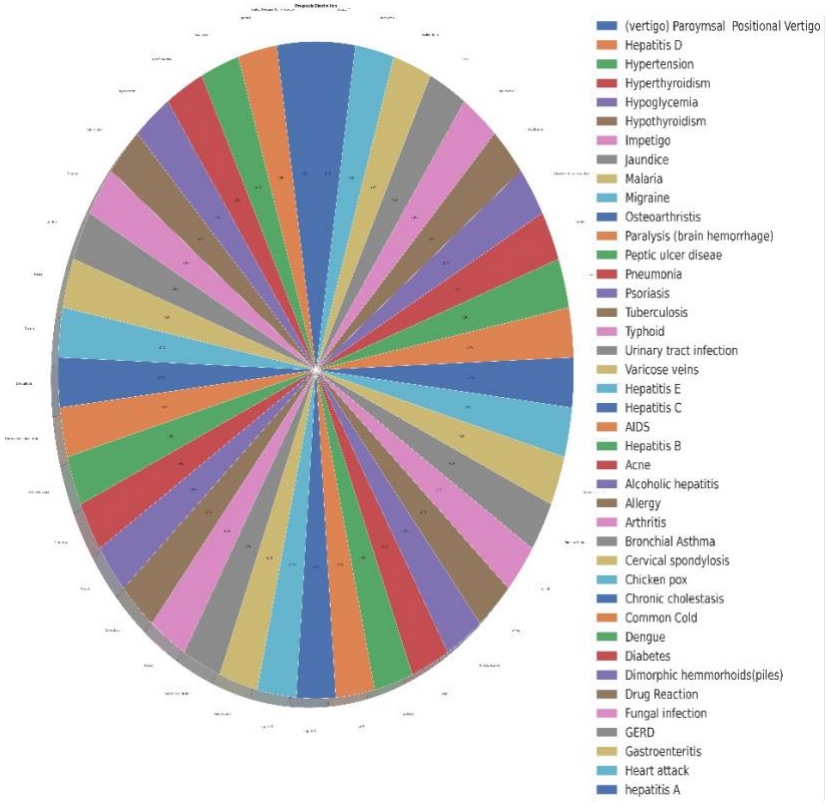

Fig:5.2.1: count of different prognosis

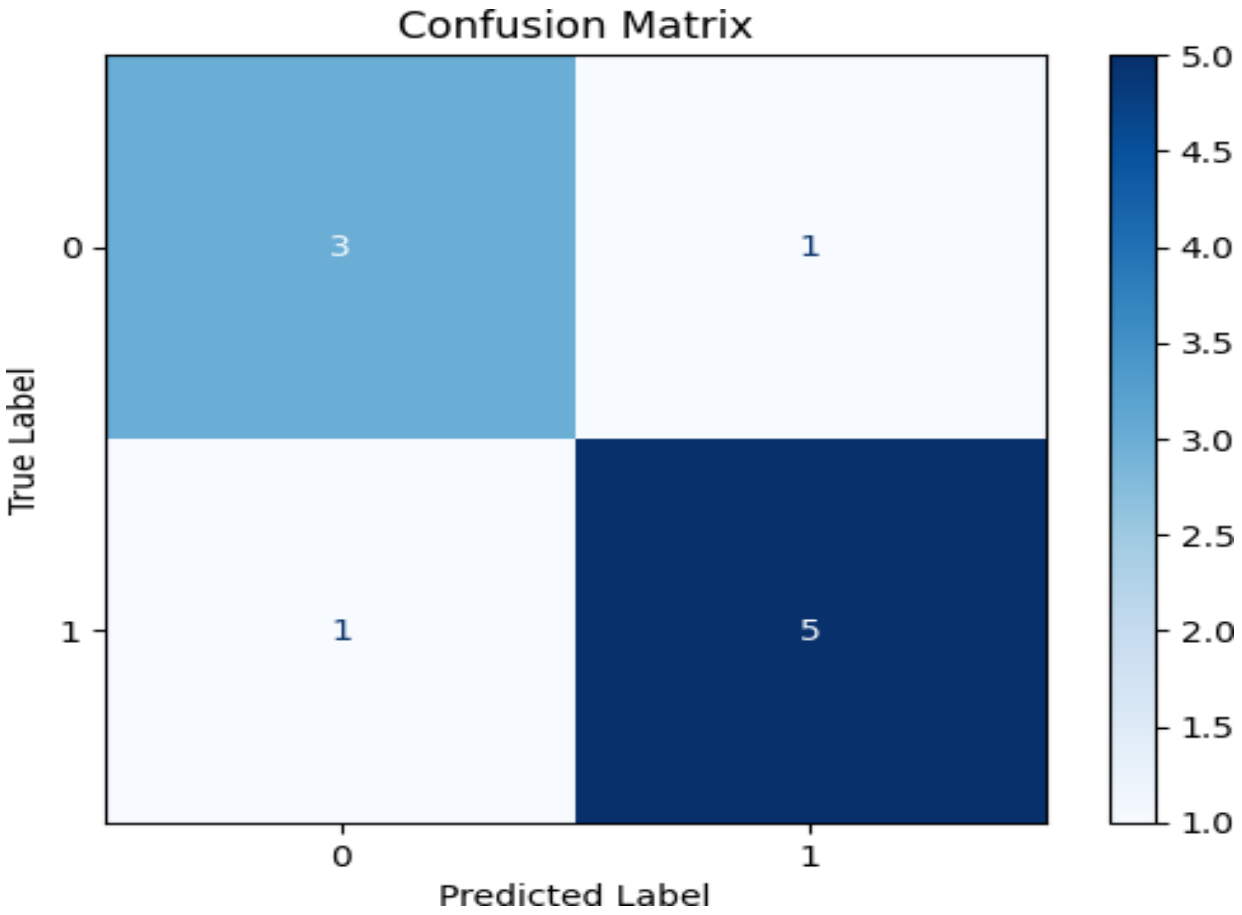
Confusion Matrix
3
1
1
5
0
1
True Label
0
1
Predicted Label
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0

Fig:5.2.2:confusion matrix

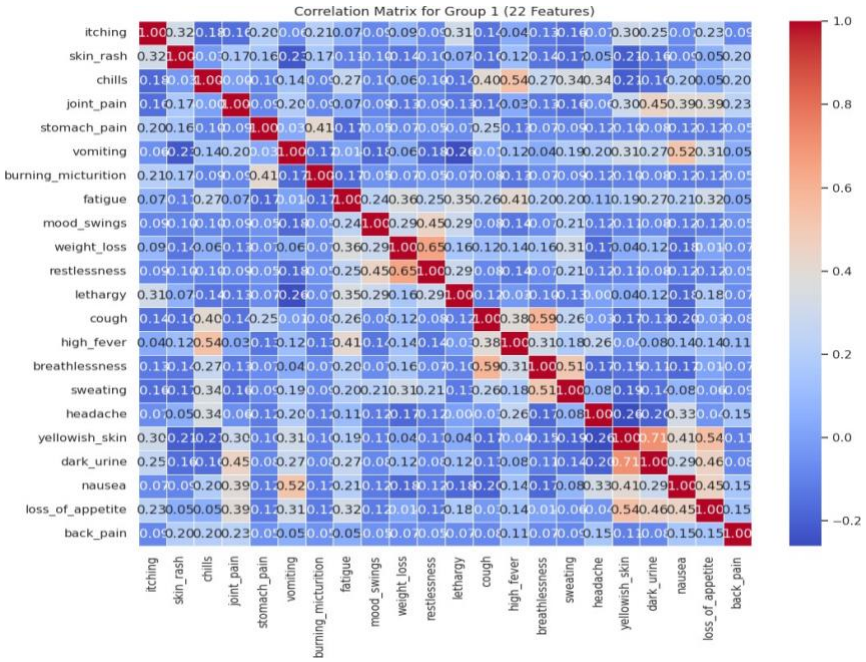

Fig:5.2.3: correlation matrix for group1 features

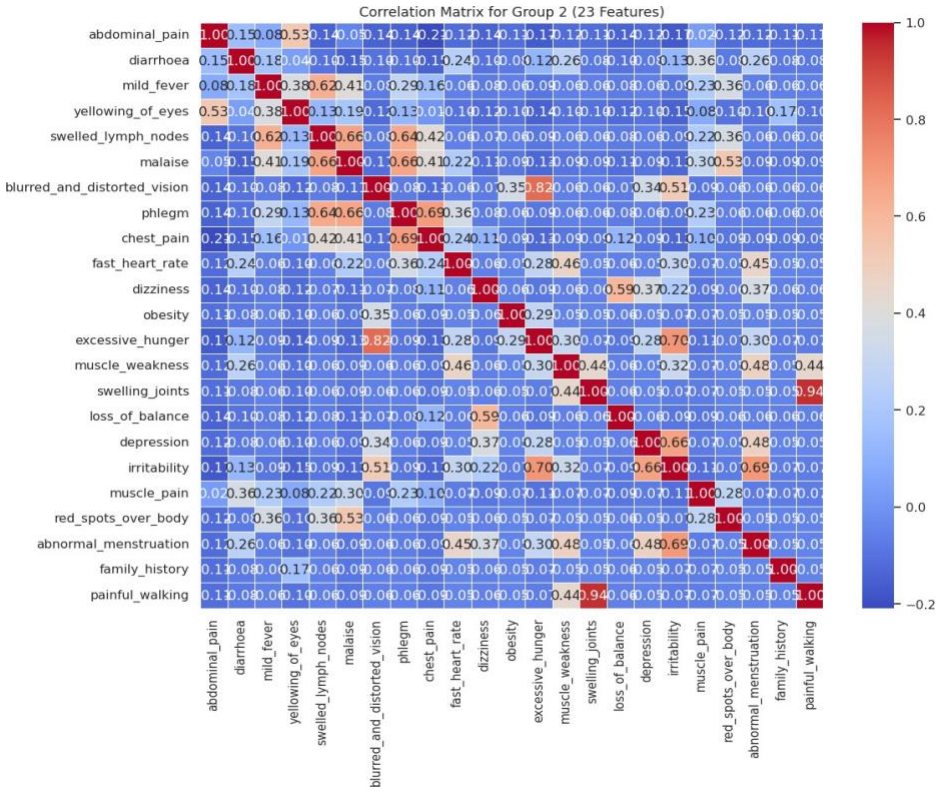

Fig:5.2.4:correlation matrix for group 2 features

RESULTS

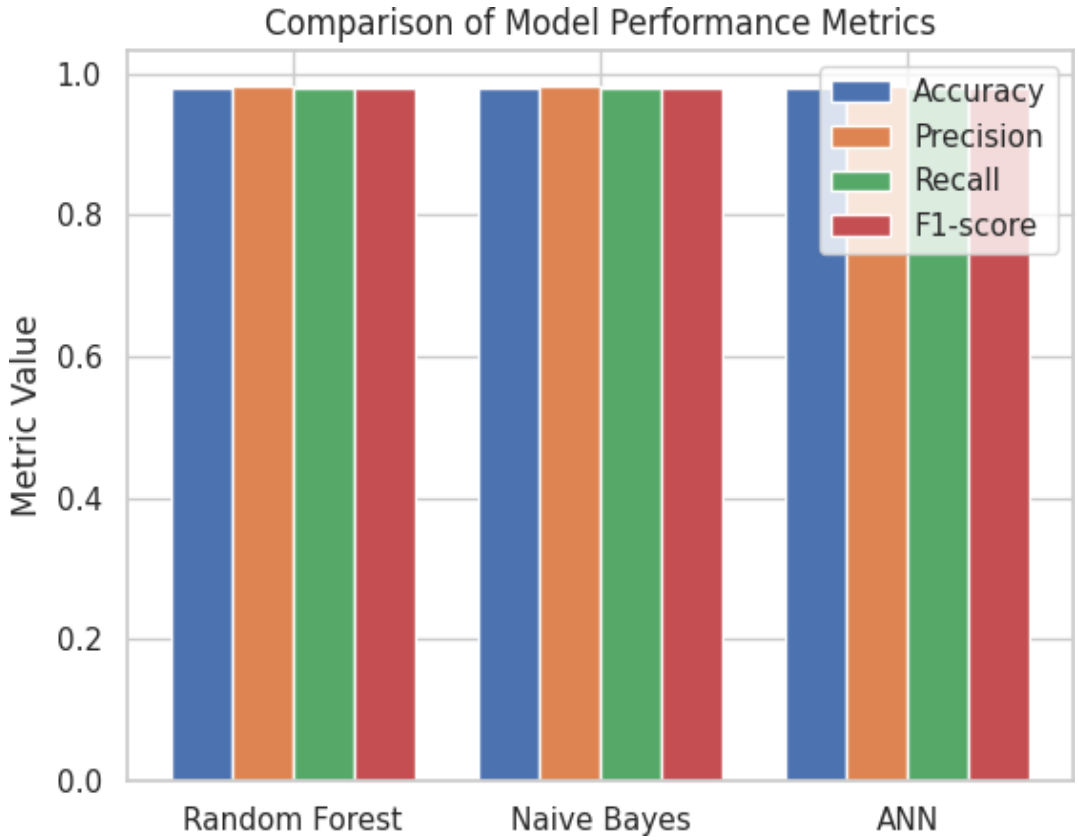

Fig:6.1: Comparision of all 3 models using different metrics

Algorithm used	Accuracy score
Random Forest	98
ANN	98
Voting Model	98

Fig:6.2 accuracy of different implemented models

From the above table, we can infer that all the algorithms have equal accuracyscore. The accuracy in terms of percentage is: 98. percentage.

6. CONCLUSION

Our work presents a comprehensive comparative study of three algorithms performance on a medical record each yielding an accuracy up to 98 percent. The main difference from previous works is that we have implemented feature selection and reduced features to 50 symptoms and we have obtained accuracy of 98 percentage. The performance is analyzed through confusion matrix and accuracy score.

7. Future Scope

Artificial Intelligence will play even more important role in data analysis in the future due to the availability of huge data produced and stored by the modern technology. Prospective research in the healthcare arena could focus on the promising field of symptom-based disease prediction. Explainable artificial intelligence (XAI) can help us make more accurate comorbidity forecasts and make the reasons behind those predictions understandable. This can enhance patient outcomes by assisting healthcare professionals in making well-informed decisions. Finding new illness connections and learning more about the mechanisms behind diseases can both be aided by the integration of genomic and electronic health record (EHR) data. This may result in early diagnosis and better care quality.

8. BIBLIOGRAPHY

[1] Grampurohit, S., & Sagarnal, C. (2020, June). Disease prediction using machine learning algorithms. In 2020 International Conference for Emerging Technology (INCET) (pp. 1-7). IEEE.

[2] Uddin, S., Khan, A., Hossain, M. E., & Moni, M. A. (2019). Comparing different supervised machine learning algorithms for disease prediction. BMC medical informatics and decision making, 19(1), 1-16.

[3] Dahiwade, D., Patle, G., & Meshram, E. (2019, March). Designing disease prediction model using machine learning approach. In 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC) (pp. 1211-1215). IEEE.

[4] Nilashi, M., bin Ibrahim, O., Ahmadi, H., & Shahmoradi, L. (2017). An analytical method for diseases prediction using machine learning techniques. Computers & Chemical Engineering, 106, 212-223.

[5] Arumugam, K., Naved, M., Shinde, P. P., Leiva-Chauca, O., Huaman-Osorio, A., & Gonzales-Yanac, T. (2023). Multiple disease prediction using Machine learning algorithms. Materials Today: Proceedings, 80, 3682-3685.

[6] Yu, Z., Wang, K., Wan, Z., Xie, S., & Lv, Z. (2023). Popular deep learning algorithms for disease prediction: a review. Cluster Computing, 26(2), 1231-1251.

[7] Yaqoob, A., Aziz, R. M., Verma, N. K., Lalwani, P., Makrariya, A., & Kumar, P. (2023). A review on nature-inspired algorithms for cancer disease prediction and classification. Mathematics, 11(5), 1081.

[8] Sun, W., Zhang, P., Wang, Z., & Li, D. (2021). Prediction of cardiovascular diseases based on machine learning. ASP Transactions on Internet of Things, 1(1), 30-35.

[9] Nemade, V., & Fegade, V. (2023). Machine Learning Techniques for Breast Cancer Prediction. Procedia Computer Science, 218, 1314-1320.

[10] Chaubey, G., Bisen, D., Arjaria, S., & Yadav, V. (2021). Thyroid disease prediction using machine learning approaches. National Academy Science Letters, 44(3), 233-238.