Complex Engineering Activity
Probability and Random Variables

Abdullah
Dept. of Computer Engineering
Air University
Islamabad
180374@students.au.edu.pk

Abdur Rehman
Dept. of Computer Engineering
Air University
Islamabad
180380@students.au.edu.pk

Muhammad Usama Ameer
Dept. of Computer Engineering
Air University
Islamabad
180369@students.au.edu.pk

***Abstract*—Disease Prediction is a system which predicts the disease based on the information or the symptoms he/she enter into the system and provides the accurate results based on that information. If the patient is not much serious and the user just wants to know the type of disease, he/she has been through. It is a system which provides the user the tips and tricks to maintain the health system of the user and it provides a way to find out the disease using this prediction. Now a day's health industry plays major role in curing the diseases of the patients so this is also some kind of help for the health industry to tell the user and also it is useful for the user in case he/she doesn't want to go to the hospital or any other clinics, so just by entering the symptoms and all other useful information the user can get to know the disease he/she is suffering from and the health industry can also get benefit from this system by just asking the symptoms from the user and entering in the system and in just few seconds they can tell the exact and up to some extent the accurate diseases. This Disease Prediction is completely done with the help of Bayes' rule and Python Programming language also using the dataset that is available previously by the hospitals using that we will predict the disease.**

I. Introduction

"Disease Prediction" system based on predictive modeling predicts the disease of the user on the basis of the symptoms that user provides as an input to the system. The system analyzes the symptoms provided by the user as input and gives the probability of the disease as an output.

Now a day's doctors are adopting many scientific technologies and methodology for both identification and diagnosing not only common disease, but also many fatal diseases. The successful treatment is always attributed by right and accurate diagnosis. Doctors may sometimes fail to take accurate decisions while diagnosing the disease of a patient, therefore disease prediction systems which use machine learning algorithms assist in such cases to get accurate results. The project disease prediction using machine learning is developed to overcome general disease in earlier stages as we all know in competitive environment of economic development the mankind has involved so much that he/she is not concerned about health according to research there are 40which leads to harmful disease later. The main reason of ignorance is laziness to consult a doctor and time concern the peoples have involved themselves so much that they have no time to take an appointment and consult the doctor which later results into fatal disease. According to research there are 70and 25project is that a user can sit at their convenient place and have a check-up of their health the UI is designed in such a simple way that everyone can easily operate on it and can have a check-up.

Disease Predictor is a system that predicts the disease of the user with respect to the symptoms given by the user. With the help of data set given to the system, Disease Predictor will be able to know the probability of the disease with the given symptoms.

As the use of internet is growing every day, people are always curious to know different new things. People always try to refer to the internet if any problem arises. People have access to internet than hospitals and doctors. People do not have immediate option when they suffer with particular disease. So, this system can be helpful to the people as they have access to internet 24 hours.

II. Methodology

Disease Predictor we have implemented is using techniques like , decision tree and Bayesian Inference algorithm. From the analysis it was found that it is one of the most accurate technique than the others.

A. Decision tree

Decision tree (DT) is one of the earliest and prominent probabilistic algorithms. A decision tree models the decision logics i.e., tests and corresponds outcomes for classifying data items into a tree-like structure.

The nodes of a DT tree normally have multiple levels where the first or top-most node is called the root node. All internal nodes (i.e., nodes having at least one child) represent tests on input variables or attributes. Depending on the test outcome, the classification algorithm branches towards the appropriate child node where the process of test and branching repeats until it reaches the leaf node . The leaf or terminal nodes correspond to the decision outcomes. DTs have been found easy to interpret and quick to learn, and are a common component to many medical diagnostic protocols . When traversing the tree for the classification of a sample, the outcomes of all tests at each node along the path will provide sufficient information to conjecture about its class. An

illustration of an DT with its elements and rules is depicted in Fig.1

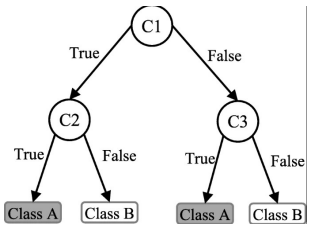

Fig. 1. A Decision Tree

B. Bayesian Inference

Bayesian inference is a method of statistical inference in which Bayes' theorem is used to update the probability for a hypothesis as more evidence or information becomes available. Bayesian inference is an important technique in statistics, and especially in mathematical statistics.

Bayesian inference has found application in a wide range of activities, including science,engineering,philosophy,medicine,sport,and law. In the philosophy of decision theory, Bayesian inference is closely related to subjective probability, often called "Bayesian probability".

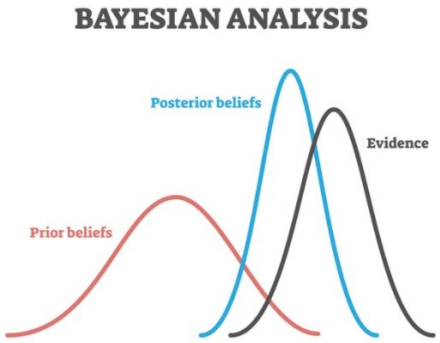

Fig. 2. Bayesian Inference

III. Background

Machine learning algorithms employ a variety of statistical, probabilistic and optimisation methods to learn from past experience and detect useful patterns from large, unstructured and complex datasets. These algorithms have a wide range of applications.

The scope of this system is primarily on the performance analysis of disease prediction approaches using different variants of supervised algorithms. Disease prediction and in a broader context, medical informative, have recently gained significant attention from the data science research community in recent years. This is primarily due to the wide adaptation of computer-based technology into the health sector in different forms (e.g., electronic health records and administrative data) and subsequent availability of large health databases for researchers. These electronic data are being utilised in a wide range of healthcare research areas such as the analysis of healthcare utilisation.

IV. System Features

The features of Disease Prediction are as follows.

- This Project will predict the diseases of the patients based on the symptoms and other general information using the datasets
- This is done based on the previous datasets of the hospitals so after comparing it can provide up to 80% of accurate results, and the project is still developing further to get the 100% accurate results.
- With the help of Disease prediction, it can predict the disease of the patient and can solve various problems and prevents from various aspects.
- The disease is predicted using the algorithms and the user has to enter the symptoms from the given drop-down menu, in order to get correct accuracy, the user has to enter all the symptoms.
- To make user more internet friendly rather than discussing with others for their disease.

V. System Design

A. Pseudo Code

- We preferred bayesian Inference to build our system.
- Bayesian inference mainly works on Bayes rule.
- Bayes' Theorem is used to calculate or update a conditional probability based on other information available. The equation is

$$P(y \mid X) = \frac{p(X \mid y)p(y)}{p(X)} \tag{1}$$

  Here X is random Variable whose value can be from x1 to xn.
- When we are working with random variables. Bayes' rule can be rewritten as:

$$P(y|X_1, X_2...X_n) = \frac{P(X_1|y)P(X_2|y)...P(X_n|y)(P(y))}{P(X_1)P(X_2)...P(X_n))} \tag{2}$$

- We will count the symptoms and then calculate their conditional probabilities.
- Calculated probabilities will be then used in Bayes's rule.
- Then Bayes' rule will give us final probability of disease.
- Now we will normalize probability by adding calculated probability of all diseases which will give us total probability.
- The probability of particular disease will be then divided by this total probability. which will normalize our probability.

B. Code Snippets

- Python Module of Bayesian inference . Make module of Bayesian Inference, which you can import in your code.

```
class BayesianClassifier:
    """This class contain all required methods for reading, cleaning
    calculating initial probabilities and predicting new incoming symptoms
    This class is the coded implementation of Bayesian Rule
    ATTRIBUTES:

    Functions:
```

Fig. 3. Class BayesianClassifier

- You can import module and use it Function. It has following Functions.(getData, CleanGivenData, train, Predict)

```
from baysianClasifier import BayesianClassifier

myClasifier = BayesianClassifier()
myClasifier.getData("../DataSet/Training.csv", ["n/a", "na", "--"], ',')
myClasifier.cleanGivenData()
myClasifier.train([132], list(range(0, 132)))
results = myClasifier.runTests([ "red_sore_around_nose"])
print(myClasifier.mostProbably(results))
```

Fig. 4. Import BayesianClassifier

- Finding NAN rows from the Data and then Drop(Delete) these rows from Dataset.

```
24 # Data cleaning of training data
25 missing_values = ["n/a", "na", "--"]
26 myData = pd.read_csv("Dataset.csv", na_values = missing_values)
27
28 FindingNAN = myData.isnull()
29
30 dropingrow = []
31
32 NAN_index = FindingNAN.any(axis=1)                    # finding index of NAN row
33
34 for i in range(0,len(myData)):
35     if(NAN_index[i] == True):
36         dropingrow.append(i)
37
38 myData = myData.drop(dropingrow)        # Drop rows which contain NAN
39 ##################################33
```

Fig. 5. Cleaning Data

- Find Probability of all Disease given symptoms and after that normalize the probabilities of all Diseases.

```
# Calculating Percentage of Symptoms

for a in range(0,F):
    sympercentageD = []
    for i in range(1, len(sym)+1):
        sympercentageD.append(D[a][i]/D[a][0])
    sympercentageD1.append(sympercentageD)              # Calculating percentage of all sympt
```

Fig. 6. Training Data

```
l = ["Fever", "Not Fever"]
E = []
E.append(D[1])
E.append(D[0]-D[1])
plt.pie(E, labels = l)
plt.show()
```

Fig. 7. Visualization Symptoms

```
# making decision what is the most probability the disease
disease = str()
lastProbability = int()

for eachDisease, diseaseProbability in probabilityResultsList.items():
    if lastProbability < diseaseProbability:
        lastProbability = diseaseProbability
        disease = eachDisease
return disease
```

Fig. 8. Predicting Data

C. State Diagram

It explains different state of the system. First the user opens Disease Predictor. The user selects the symptoms. When finished selecting symptoms the user submits the symptoms. Disease Predictor analyzes the symptoms and displays the results.

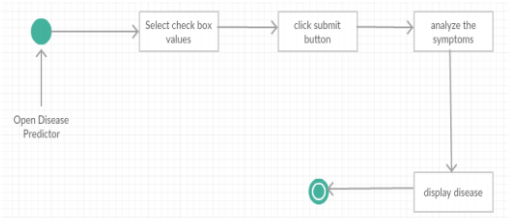

Fig. 9. State Diagram

D. Sequence Diagram

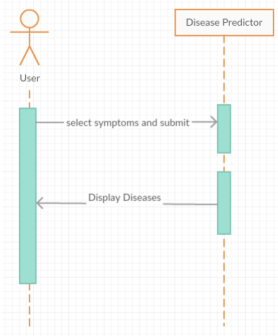

Fig. 10. Sequence Diagram

It explains the sequence of the Disease Predictor. Initially system shows the symptoms to be selected. The user selects the symptoms and submits to the system .The Disease Predictor predicts and display the results.

VI. Results

- We gave some of the symptoms to the system to predict disease according to our algorithm.
- After processing the system gave us the probability of all the diseases and the disease with most probability.

Fig. 11. Input of Symptoms

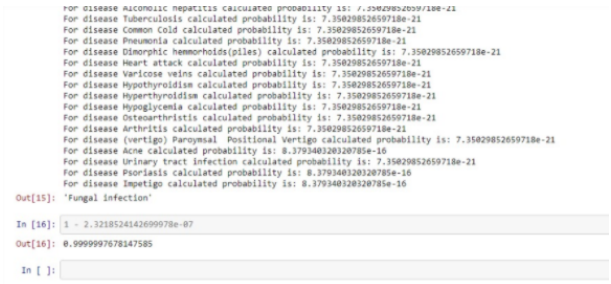

Fig. 12. Disease Probability

VII. Command Line Output

We Will run CEP _ dict_ implementation.py by typing
python [filename] [testing_ data.csv] [Row number]
or
python CEP_dict_implementation.py ../DataSet/Testing.csv 5.

Program will read test.csv and read specific row then find all the symptoms of that row and train all model then predict disease.

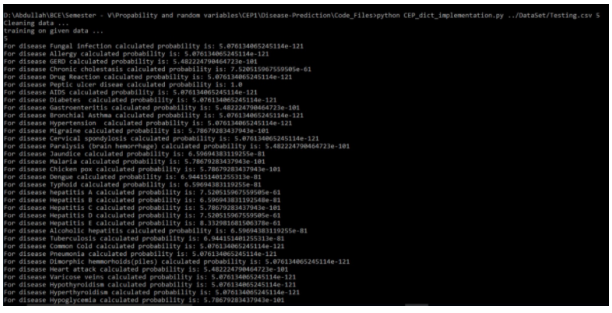

Fig. 13. Command Line Output

VIII. Future Enhancement

- Interactive user interface.
- Facility for modifying user detail.
- Can be done as Web page.
- More Details and Latest Diseases.
- Can be done as Mobile Application.

IX. Conclusions

This project aims to predict the disease on the basis of the symptoms. The project is designed in such a way that the system takes symptoms from the user as input and produces output i.e. predict disease. Total diseases included were 41 with 133 different symptoms. Average prediction accuracy probability is very satisfactory.

Our proposed system helps to improve the accuracy of diagnosis and greatly helpful for further treatment. In future enhancements, the accuracy has to be tested with different dataset and to apply other AI algorithms to check the accuracy estimation.

In this work, we show that Automated Disease Prediction System can help people who are facing difficulties, better understand their physical condition by predicting potential diseases. We also show that our framework enables the system to perform significantly better than existing ones. Having said that, our system accuracy can be increased further as there is space left for improvement. Like the decision tree and parent tree generation is a cumbersome task but it is a continuous process, same goes with the enrichment of the database. It will get better and better over time and accuracy of disease prediction will also be on the rise.

X. Recommendations

This project has not implemented recommendation of medications to the user. So, medication recommendation can be implemented in the project. History about the disease for a user can be kept as a log and recommendation can be implemented for medications.

XI. Future Work

Every one of us would like to have a good medical care system and physicians are expected to be medical experts and take good decisions all the time. But it's highly unlikely to memorize all the knowledge, patient history, records needed for every situation. Although they have all the massive amount of data and information; it's difficult to compare and analyse the symptoms of all the diseases and predict the outcome.

So, integrating information into patient's personalized profile and performing an in-depth research is beyond the scope a physician. So the solution is ever heard of a personalized healthcare plan – exclusively crafted for an individual. Predictive analytics is the process to make predictions about the future by analyzing historical data. For health care, it would be convenient to make best decisions in case of every individual. Predictive modeling uses artificial intelligence to create a prediction from past records, trends, individuals, diseases and the model is deployed so that a new individual can get a prediction instantly. Health and Medicare units can use these predictive models to accurately assess when a patient can safely be released.

XII. Acknowledgement

we express our sincere gratitude towards our guide of Prof. Dr. Mudassar Farooq for his constant help, encouragement and inspiration throughout the project work. Also we would like to thank him for valuable guidance , ability to motive us and even willingness to solve difficulty made it possible to make our project unique and made task easier.

XIII. References

- https://www.kaggle.com/neelima98/disease-prediction-using-machine-learning
- https://www.machinelearningplus.com/predictive-modeling/how-naive-bayes-algorithm-works-with-example-and-full-code/
- https://www.geeksforgeeks.org/naive-bayes-classifiers/