

Disease Prediction: Various Symptoms Using Machine learning

Ms. Meghna Singh¹, Ashish Richhariya², Brijesh Gupta³

¹Department of Engineering Science & Humanities, Thakur College Of Engineering and Technology, Mumbai, India

²Department of Film Production and Communication, KES Shroff College, Mumbai, India

³Department of Engineering Science & Humanities, Thakur College Of Engineering and Technology, Mumbai, India

Abstract:

In recent years, the use of machine learning algorithms has become popular in the healthcare industry for predicting diseases. In this paper, we propose a framework for disease prediction that utilizes three popular algorithms, Decision Tree, Random Forest Tree, and Naive Bayes. We have outlined disease prediction framework utilizing different ML Calculations. The dataset utilized had more than 230 maladies for processing. Based on the side effects, age, sexual orientation of an individual, the conclusion framework gives the yield as the disease that the person may well be enduring from. The weighted Decision Tree calculation gave the finest comes about as compared to the other calculations. The exactness of the weighted Decision Tree calculation for the forecast was 95.17%. Other algorithms i.e. Random Forest Tree and Naive Bayes also gave the exactness of 95%. If a recommendation system can be made for doctors and medicine while using review mining will save a lot of time. In this type of system, the user face problem in understanding the heterogeneous medical vocabulary as the users are laymen. User is confused because a large amount of medical information on different mediums are available. The idea behind this system is to adapt with the special requirements of the health domain related with users.

Keywords: Decision Tree, Random Forest Tree, Navie Bayes, Exactness

I. INTRODUCTION

The World Health Organization (WHO) records cardiovascular diseases as the driving cause of passing all-inclusive with 17.9 million individuals dying each year. Disease prediction is a crucial task in the healthcare industry as it enables early detection of diseases and facilitates better treatment and management. Machine learning algorithms have shown great potential in predicting diseases, and various studies have been conducted in this field. In this paper, we propose a disease prediction framework that utilizes three popular algorithms, Decision Tree, Random Forest Tree, and Naive Bayes. The chance of heart infection increment due to destructive behavior that leads to overweight and weight, hypertension, hyper glycaemia, and cholesterol. Besides, the American Heart Association complements side effects with weight pick up (1–2 kg per day), sleep issues leg swelling, inveterate hack and tall heartrate. Diagnosis may be an issue for professionals due the symptoms' nature of being common to other conditions or confounded with signs of maturing. In later along time professional have expanded their utilization of computer advances to makes trades decision making back. Within-the health care

industry, machine learning is getting to imperative arrangement to help the conclusion of patients. Machine learning is an analytical tool used when a task is large and difficult to program, such as transforming medical record into knowledge, pandemic predictions, and genomic data analysis [1]. There are too as many further municipalities which need restorative installations. Virtual specialists are board-certified specialists who choose to hone online through videotape and phone movables, rather of in-person arrangements but this isn't conceivable within the case of extremity. Machines are always considered superior than people as, without any mortal mistake, they can perform errands more efficiently and with a steady position of fineness. A sickness predictor can be called a virtual specialist, which can prognosticate the illness of any patient without any mortal error also, in conditions like COVID-19 and EBOLA, an infection index can be a favoring because it can identify a human's sickness without any physical contact (2) (3). Hence a system was needed that can prognosticate any complaint grounded on the symptoms with a good chance of delicacy at least ranging from 90-95 fineness. Then in this system we're going to use 3 machine learning algorithms Decision Tree, Random Forest Tree and Naive Bayes (Considering the result of the algorithms with loftiest number of fineness). Different studies have shown promising results while using machine learning algorithms (3).

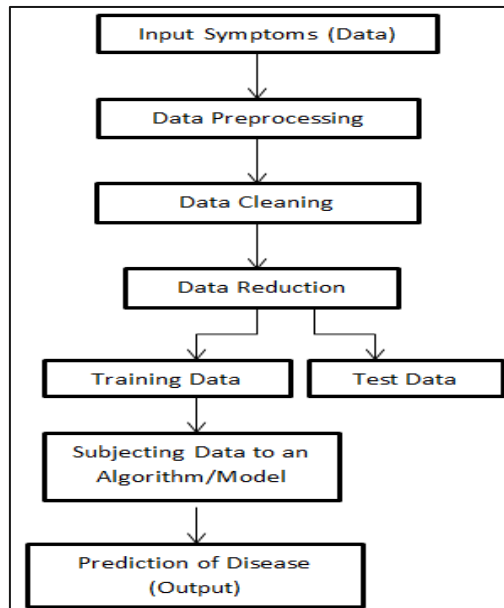
II. LITERATURE REVIEW

The viability and security of zanamivir, managed 2x or 4x day by day over 5 days, was estimated in the treatment of flu conditions. An add up to of 1256 cases entered the consider; 57 of those randomized had laboratory-verified flu impurity. The essential conclusion point, "relief of major symptoms," was made to assess contrasts in clinical affect. In the overall crowd with or without flu impurity, zanamivir dropped the middle number of days to reach this conclusion point by 1 day. The drop was more prominent in cases treated inside 30 h of suggestion onset, febrile at suppose about section, and in characterized high-threat bunches. Zanamivir dropped gloamings of disturbed sleep, time to resumption of typical exercises, and use of suggestion relief drugs. It was well endured. These comes about propose that zanamivir can basically dwindle the length and overall characteristic impact of flu [1]. In this composition, sample datasets of 5145 cases, including 686 laboratory test results were collected where an aggregate of 39 specific conditions grounded on the International Classification of conditions, 10th modification (ICD-10) canons were delved. These datasets were used to construct light grade boosting machine (LightGBM) and extreme grade boosting (Boost) models and a DNN model using Tensor Flow. The optimized ensemble model achieved an F1-score of 81 and vaccination delicacy of 92 for the five most common conditions. The deep learning and ML models showed differences in prophetic power and complaint classification patterns [2].

The healthcare assiduity has set up that Machine learning (ML) is a useful and accurate decision-making fashion in the data collection produced in large amounts. The medical decision support systems developed were effective grounded on the software and the different algorithms proposed by numerous experimenters. Then a study is done grounded on the colorful ways using the different algorithms and their performance analysis. The prognosticating model was introduced with several combined features, and among the multiple styles and were other classification techniques. numerous being ways banded, among which the delicacy position was set up as 88.7 using the Hybrid Random Forest with a Linear Model (HRFLM) fashion.

III. METHODOLOGY FLOW

The dataset we have considered comprises of 132 indications, the blend or stages of which leads to 41 illnesses. In light of the 4920 documents of various patient samples, mainly to point foster a forecast algorithm that considers in the side. The dataset we have considered comprises of 132 indications, the blend or stages of which leads to 41 illnesses. In light of the 4920 documents of various patient samples, mainly to point foster a forecast algorithm that considers in the side effects of various client and forecasts the sickness that the person is bound to be affected.



Inputs (Patient Symptoms): When planning the algorithm, we have expected to be the client can have an unmistakable thought regarding the indications he is encountering. The Prediction created considers 95 manifestations in the midst of which the client can permit the indications his preparing as the input

	Disease	Count of Disease Occurrence	Symptom
0	UMLS:C0020538_hypertensive disease	3363.0	UMLS:C0008031_pain chest
1	UMLS:C0020538_hypertensive disease	3363.0	UMLS:C0392680_shortness of breath
2	UMLS:C0020538_hypertensive disease	3363.0	UMLS:C0012833_dizziness
3	UMLS:C0020538_hypertensive disease	3363.0	UMLS:C0004093_asthenia
4	UMLS:C0020538_hypertensive disease	3363.0	UMLS:C0085639_fall

Fig 2. Input data

	disease	symptom	occurrence_count
0	hypertensive disease	shortness of breath	3363.0
1	hypertensive disease	dizziness	3363.0
2	hypertensive disease	asthenia	3363.0
3	hypertensive disease	fall	3363.0
4	hypertensive disease	syncope	3363.0

Fig 3. Preprocess

B) Data pre-processing: The mining of the data's approaches that changes the crude information or then again encrypts the information to form a structure so that it can be effectively deciphered with the help of calculation is known as information pre-processing. The information pre-processing strategies utilized in the introduced work which listed as follows:

1) Data Purification: Data is purified using certain measures like stuffing in lost worth, along these lines settling the irregularities in the information.

Data Reduction: The examination turns out to be hard when managing a gigantic information base. Thus, we kill those autonomous variables (symptoms) which may not affect the objective variables (diseases). So in the progress task, of around 95 of 132 side effects firmly identified with the illnesses are chosen

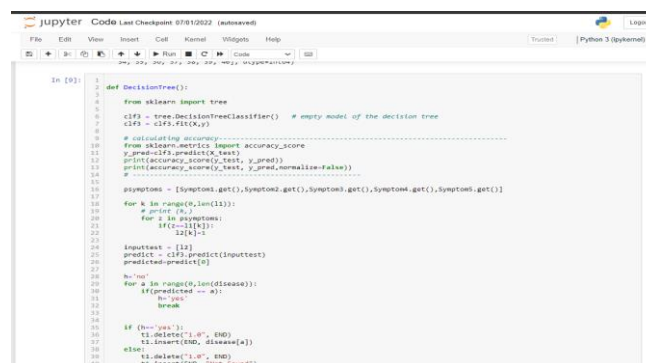
C) Models: The entire system is designed in such a way to predict the diseases by utilizing the three Algorithms i.e., Decision Tree model, Naive Bayes model and Random Forest classifier model, so that the predictive analysis study is proposed at the end of the study by exploring its speed, efficiency and performance of the various algorithms for the input dataset.

D) Output(diseases): While a framework is made with the preparation set utilizing the validated calculations standard datasets are shaped and whenever the client indications are provided as a contribution as input of the algorithm, and the side effects are composed agreeing as the standard dataset created, accordingly creating arrangements and foreseeing the high probable infection.

IV.WORKING

A) The Disease forecast framework is executed utilizing the three information mining calculations for example Random Forest, Decision tree classifier and Naive Bayes. The portrayal and implementation of the calculations are provided.

B) Decision Tree Model: The order of the algorithm worked as Decision tree look like the model of many branches in a tree. So, by analyzing the arrangement of unequivocal assuming at that point rules on highlight esteems (manifestations for our situation), it classifies down the dataset into more modest and more modest subsets that outcomes in anticipating an objective value (disease). A tree comprises of the mainly a decision Node and a Leaf Node. Decision Node: It has a minimum of 2 branches. In our analysis we introduced, every one of the manifestations are taken as decision node. Leaf Node: Constitutes the order which denotes that, the decision may of any of the branch. So that the diseases here represent to a leaf node



```
In [9]: def DecisionTree():
1   from sklearn import tree
2   clf = tree.DecisionTreeClassifier() # empty model of the decision tree
3   clf = clf.fit(X,y)
4
5   # calculating accuracy
6   from sklearn.metrics import accuracy_score
7   y_pred=clf.predict(X_test)
8   print(accuracy_score(y_test, y_pred))
9   print(accuracy_score(y_test, y_pred,normalize=False))
10
11 #
12
13 symptoms = [Symptom1.get(),Symptom2.get(),Symptom3.get(),Symptom4.get(),Symptom5.get()]
14
15 for k in range(0, len(X)):
16     # print (X, y)
17     for i in symptoms:
18         if (i==X[k]):
19             print(i)
20
21
22
23 inputtext = []
24 predict = clf.predict(inputtext)
25 predicted=predict[0]
26
27 h=""
28 for a in range(0, len(disease)):
29     if(predicted == a):
30         h=""
31         break
32
33
34 if (h=="yes"):
35     ti.delete(1,0", END)
36     ti.insert(END, disease[a])
37 else:
38     ti.delete(1,0", END)
39     ti.insert(END, "Not Found")
```

Fig 4: Code Snapshot of Decision Tree

C) Random Forest Classifier: The Random Forest classifier is adaptable, and simple to utilize AI calculation that gives remarkable outcomes more often than not applied without any hyper tuning. So, as

validated in the Decision tree model, the notable restriction of tree calculation is overfitting. So, it shows up as though the decision tree has remembered core of the information. This model forestalls this issue: That It's a form of troupe investigation. Troupe investigation alludes to utilizing different calculations or identical calculation on numerous occasions. Random Forest model is a group of many decision trees. Also, more noteworthy the quantity of these trees in this model is the fitter of the speculation.

All the more decisively, the random Forest fills in as listed below:

- 1) Fix the 'k' side effects from data (clinical data) the sum of m manifestations arbitrarily (here $k \ll m$). At that point, it assembles a Decision tree model with the help of side effects of 'k'.
- 2) Rehashes 'n' number of times with the goal that we have n number of tree model worked from various Random mixes of indications which is denoted as 'k' (or an alternate irregular example of information, known as bootstrap test).
- 3) Consider every one of the various n-constructed trees and proceeds a variable which is random to foresee the illness. Here it Store the anticipated illness, so that we can have a sum of 'n' illness anticipated from n number of the decision tree model.
- 4) Computes the decisions in favor of each anticipated illness and consider the mode (which is most continuous illness anticipated) as last expectation from the Random Forest model calculation.

```
In [10]: 1 def randomforest():
2         from sklearn.ensemble import RandomForestClassifier
3         clf4 = RandomForestClassifier()
4         clf4 = clf4.fit(X,np.ravel(y))
5
6         # calculating accuracy-----
7         from sklearn.metrics import accuracy_score
8         y_pred=clf4.predict(X_test)
9         print(accuracy_score(y_test, y_pred))
10        print(accuracy_score(y_test, y_pred,normalize=False))
11        # -----
12
13        psymptoms = [Symptom1.get(),Symptom2.get(),Symptom3.get(),Symptom4.get(),Symptom5.get()]
14
15        for k in range(0,len(11)):
16            for i in psymptoms:
17                if(i==11[k]):
18                    l1[k]=1
19
20        Inputtest = [12]
21        predict = clf4.predict(inputtest)
22        predicted=predict[0]
23
24        h="no"
25        for a in range(0,len(disease)):
26            if(predicted == a):
27                h="yes"
28                break
29
30        if (h=="yes"):
31            t2.delete("1.0", END)
32            t2.insert(END, disease[a])
33        else:
34            t2.delete("1.0", END)
35            t2.insert(END, "Not Found")
36
```

Fig 5: Coding Snapshot of Random Forest Classifier

A) Naïve Bayes Classifier Algorithm: Naive Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem. Maximum-likelihood training can be done by evaluating a closed-form expression which takes linear time rather than by expensive iterative approximation as used for many other types of classifiers. In the statistics literature, naive Bayes models are known under a variety of names, including simple, Bayes and independent Bayes. All these names reference the use of Bayes' theorem in the classifier's decision rule, but naive Bayes is not (necessarily) a Bayesian method.

```
In [11]: 1 def NaiveBayes():
2         from sklearn.naive_bayes import GaussianNB
3         gnb = GaussianNB()
4         gnb.fit(X, y.ravel())
5
6         # calculating accuracy.....
7         from sklearn.metrics import accuracy_score
8         y_pred=gnb.predict(X_test)
9         print(accuracy_score(y_test, y_pred))
10        print(accuracy_score(y_test, y_pred,normalize=False))
11        # .....
12
13        psymptoms = [Symptom1.get(),Symptom2.get(),Symptom3.get(),Symptom4.get(),Symptom5.get()]
14        for k in range(0,len(11)):
15            for z in psymptoms:
16                if(z==11[k]):
17                    12[k]=1
18
19        inputtest = [12]
20        predict = gnb.predict(inputtest)
21        predicted=predict[0]
22
23        h='no'
24        for a in range(0,len(disease)):
25            if(predicted == a):
26                h='yes'
27                break
28
29        if (h=='yes'):
30            t3.delete("1,0", END)
31            t3.insert(END, disease[a])
32        else:
33            t3.delete("1,0", END)
34            t3.insert(END, "Not Found")
35
```

Fig 6: Coding Snapshot of Naïve Bayes Classifier

V. CONCLUSION

Following is the accuracy score of all the 3 algorithm.

```
0.9512195121951219
39
0.9512195121951219
39
0.9512195121951219
39
```

Fig 7: Exactness of all three algorithm

In the above figure, 0.95121 represent the accuracy score and 39 represents the disease.

```
In [10]: 1 from sklearn import metrics
2         from sklearn.tree import DecisionTreeClassifier
3         clf=DecisionTreeClassifier()
4         clf= clf.fit(X,y)
5         y_pred=clf.predict(X_test)
6         y_pred
7
8
Out[10]: array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
17, 18, 19, 20, 21,  3, 23, 24, 25, 26, 27, 28,  2, 30, 31, 32, 33,
34, 35, 36, 37, 38, 39, 40], dtype=int64)

In [11]: 1 print("Accuracy:", metrics.accuracy_score(y_test,y_pred))

Accuracy: 0.9512195121951219
```

REFERENCES

1. Anna Karen Gárate-Escamila, Amir Hajjam El Hassani, Emmanuel Andrès, Classification models for heart disease prediction using feature selection and PCA, Informatics in Medicine Unlocked, 2020.
2. Keniya, Rinkal and Khakharia, Aman and Shah, Vruddhi and Gada, Vrushabh and Manjalkar, Ruchi and Thaker, Tirth and Warang, Mahesh and Mehendale, Ninad and Mehendale, Ninad, Disease

Prediction From Various Symptoms Using Machine Learning (July 27, 2020)

3. Symptoms Based Multiple Disease Prediction Model using Machine Learning Approach Talasila Bhanuteja, Kilaru Venkata Narendra Kumar, Kolli Sai Poornachand, Chennupati Ashish, Poonati Anudeep. (IJITEE), Volume-10 Issue-9, July 2021
4. [A. S. Monto, D. M. Fleming, D. Henry, R. de Groot, M. Makela, T. Klein, M. Elliott, O. N. Keene, C. Y. ManThe Journal of Infectious Diseases, Volume 180, Published: 01 August 1999.
5. Park, D.J., Park, M.W., Lee, H. et al. Development of machine learning model for diagnostic disease prediction based on laboratory tests. Sci Rep 11, 7567 (2021). Published-07 April 2021
6. A comparative study on machine learning based heart disease prediction A. Kondababu, V. Siddhartha, BHK.