Disease Detection and Consultation using Django and Machine Learning

Nishant Mishra[1] Sujal Soumi[2] Somyajit Karmakar[3] Devraj Gorai[4]
Priyanshu Kumar[5] Aditya Kora[6], Asansol Engineering College, India

ABSTRACT

The design and implementation of various well-known data mining techniques in a variety of real-world applications (e.g., industry, healthcare, and bioscience) has led to their use in machine learning environments to extract meaningful information from provided data in healthcare communities, biological disciplines, and other fields. Early illness prediction, patient treatment, and community services all benefit from precise medical database analysis. Machine learning techniques have been effectively used in a variety of applications, including disease prediction. The goal of developing a classifier system utilising machine learning algorithms is to greatly assist physicians in predicting and diagnosing diseases at an early stage, which will greatly aid in solving health-related difficulties. For our study, a sample of 4920 patient records diagnosed with 41 disorders was chosen. We chose 95 out of 132 independent variables (symptoms) that are strongly associated to illnesses and improved them. The disease prediction system built utilising Machine learning techniques such as Decision Tree classifier, Random forest classifier, and Nave Bayes classifier is demonstrated in this research paper. This paper ***"DISEASE PREDICTION USING DJANGO AND MACHINE LEARNING"*** gives a comparison of the outcomes of the aforementioned algorithms.

Key Words
Machine Learning, Data mining, Decision Tree classifier, Random forest classifier, Naive Bayes classifier.

1. OBJECTIVE

The purpose of this research is to see if the premise that supervised machine learning algorithms may enhance health care by detecting illnesses accurately and early is true. In this paper, we look into research that use several supervised machine learning models for each illness detection issue. The system is meant to use intelligent decision tree data processing technique to guess the foremost accurate illness supported patient's symptoms. Many symptoms are fed into the system, and hence diseases associated with it. The user describes his or her symptoms and looks for further ones. As a result, the algorithm checks the database, extracts the data, and forecasts the precise disease that the person is suffering from. Several techniques will be tested for illness identification, including DT, RF, GB, KNN and GNB. The most important aspect of this strategy is that it provides many symptom possibilities so that the patient may search for any conceivable symptom. As a result, prediction accuracy improves.

After performing feature selection, the top performing ML models for each illness will be determined at the conclusion of this literature and would be used for building the desired machine learning model that would predict the disease a patient is suffering from and furthermore provide online consultation with the concerned Doctor.

2. OVERVIEW

The dataset we studied at has 132 symptoms, which may be combined or permuted to produce 41 disorders. We aim to build a prediction model based on the 4920 patient data that takes the user's symptoms and forecasts the ailment he is most likely to have.

The considered symptoms are:

TABLE I. SYMPTOMS

Symptoms		
Back pain	Bloody stool	scurrying
Constipation	depression	Passage of gases
Abdominal pain	Irritation in anus	Weakness in limbs
diarrhoea	Neck pain	Fast heart rate
Mild fever	dizziness	Internal itching
Yellow urine	cramps	Toxic look
Yellowing of eyes	bruising	palpitations
Fluid overload	Swollen legs	Prominent veins on calf
Swelling of stomach	irritability	Fluid overload
Swelled lymph nodes	Swollen blood vessels	Excessive hunger
malaise	Muscle pain	Black heads
Blurred and distorted vision	Pain in anal region	Pain during bowel movements
phlegm	Brittle nails	Rusty sputum
Throat irritation	Belly pain	Mucoid sputum
Redness of eyes	Enlarged thyroid	Puffy face and eyes
Sinus pressure	Slurred speech	Hip joint pain
Runny nose	Knee pain	polyuria
congestion	Skin peeling	Family history
Chest pain	Extra marital contacts	Swollen extremities
Yellow crust ooze	Swelling joints	Coma
Loss of smell	Stiff neck	Unsteadiness
Movement stiffness	Muscle weakness	Drying and tingling lips
Spinning movements	Red sore around nose	Weakness of one body side
Bladder discomfort	Foul smell of urine	Continuous feel of urine
Altered sensorium	Red spots over body	Abnormal menstruation
Dyschromic patches	Watering from eyes	Increases appetite
Lack of concentration	Blood in sputum	Receiving blood transfusion
Receiving unsterile injections	Blood in sputum	History of alcohol consumption
Puss filled pimples	Blood in sputum	History of alcohol consumption
Silver like dusting	Small dents in nails	Inflammatory nails
blister		

The diseases considered are:

TABLE II. DISEASES

Diseases		
Fungal Infection	Malaria	Varicose veins
Allergy	Chickenpox	Hypothyroidism

Gerd	Dengue	Vertigo
Chronic cholestasis	Peptic ulcer disease	acne
Drug reaction	Hepatitis A	Urinary tract infection
Piles	Hepatitis B	Psoriasis
AIDS	Hepatitis C	Impetigo
Diabetes	Hepatitis D	Hyperthyroidism
Gastroenteritis	Hepatitis E	Hypoglycaemia
Bronchial Asthma	Alcoholic hepatitis	Cervical Spondylosis
Hypertension	Tuberculosis	Arthritis
Migraine	Common cold	Osteoarthritis
Paralysis	Pneumonia	Typhoid
Jaundice	Heart Attack	

The generalized prediction model can be given as:

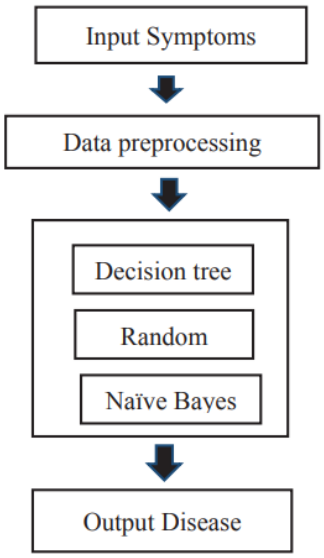


Fig. 2.1 Prediction Model

A. Input (Symptoms)

We assumed that the user had a good understanding of the symptoms he is experiencing while building the model. The generated prediction takes into account 95 symptoms, with the user providing his or her processing as input.

B. Data pre-processing

Data pre-processing is a data mining approach that transforms or encodes raw data into a form that may be easily processed by an algorithm. The pre-processing 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: When working with a large database, analysis becomes difficult. As a result, we rule out any independent variables (symptoms) that may have a minor or no effect on the target variable (disease). In this study, 95 of 132 symptoms that are closely associated to diseases are chosen.

C. Models selected

The system is trained to predict the diseases using three algorithms

- Decision Tree
- Random forest
- Gradient Boosting
- KNN
- Naïve Bayes

At the conclusion of the work, a comparison analysis is given, comparing the performance of each method in the database under consideration.

D. 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 to the rule set developed, thus making classifications and predicting the most likely disease.

3. MATERIAL / METHODS/ TOOLS:

3.1 Technologies to be incorporated and worked upon:

1. Machine Learning (Predicting kind of disease patient is suffering from)
2. Web Development (Frontend and Backend)

3.2 Languages and frameworks used:

For Machine Learning :

- Python
- Scikit Learn

For Frontend and Backend Web Interface:

- Django - Python Open-source Web Framework
- HTML
- CSS
- BOOTSTRAP - Open Source CSS Framework
- jQuery - JavaScript Library
- Flask
- Jinja
- Werkzeug

For Database Storage:

- PostgreSQL

4. PROPOSED WORK : DISEASE DETECTION AND CONSULTATION SYSTEM

We propose a real-time role-based access web-Interface that would first enable the user to Register/login as a doctor or a patient, then further on incase of Patient would provide with the choice to input his/her symptoms and would predict what disease the patient is suffering from and give a facility for online consultation to the suggested doctor for that disease.

In case of a Doctor, the user would login and get access to the consultation History and can view the patient's profile and provide real time consultation.

4.1 Machine Learning

Detection of disease from a set of given input symptoms is done through Machine Learning. Data collection has been done from the internet to identify the disease here the real symptoms of the disease are collected i.e., no dummy values are entered. The symptoms of the disease are collected from Kaggle.com and different health-related websites. This csv file contains 4920 rows of records of the patients with their symptoms (132 types of different symptoms) and their corresponding disease (41 classes of general disease). The system is trained to predict the diseases using five algorithms

- Decision Trees Classifier
- Random Forest Classifier
- Gradient Boosting Classifier
- K-nearest Classifier
- Gaussian Naive Bayes Classifier

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 to the rule set developed, thus making classifications, and predicting the most likely disease that the user is suffering from and provide online consultation.

4.2 UI-Design

Starting with the Home page, there would be five template HTML files to be used for the website – admin, consultation, Doctor, patient, and sign-in page. After the successful login of the user, sign-in is loaded where the user gets to sign-in as either Doctor, Patient or Admin. After successfully logging in into the concerned role, the user gets more options to explore, in case of patient - getting consultation facility and incase of Doctor - viewing profile of patient and consultation History menu. It is an interactive web page with responsive buttons and links and various resources such as Google font, Bootstrap, Owl Carousel, Magnific popup, CSS and Awesome Icon. The web application also takes into use static files such as javascript and CSS, supporting the display of a web page. Usually, the web server is configured to serve them for us, but during the development, these files are served from the static folder in your package or next to your module and it will be available at /static on the application. A special endpoint 'static' is used to generate URL for static files.

4.3 Django as Framework

The Interface is built upon Django which is a high-level Python web framework that encourages rapid development and clean, pragmatic design. In Django, every web app you want to create is called a project; and a project is a sum of applications. An application is a set of code files relying on the MVT pattern. As example , let's say we want to build a website, the website is our project and the forum, news, contact engine are applications. This structure makes it easier to move an application between projects since every application is independent.

Every web-app in Django has the following structure -

```
myproject/
    manage.py
    myproject/
        __init__.py
        settings.py
        urls.py
        wsgi.py
```

Fig 4.1 Django Project Structure

We have the following web apps in the system besides the main app:

- accounts: This manages the register/login of the user.
- chats: This web app handles the chat application used during the consultation between patient and the Doctor.
- disease prediction: This web app handles the password validation and authentication.

After having the patient input the symptoms, we ran our machine learning model to predict what kind of disease the patient is suffering from and from thereon provide consultation facility to the patient.

4.4 PostgreSQL as Database Storage

We save the login details of patients and Doctors using the PostgreSQL, which is an advanced, enterprise class open source relational database that supports both SQL (relational) and JSON (non-relational) querying. We first installed pgAdmin4 then clicking onto the server, we created our database Predico. Then we get the database config part as,

DATABASES = {

```
    'default': {
        'ENGINE': 'django.db.backends.postgresql',
        'NAME': 'predico',
        'USER': 'postgres',
        'PASSWORD': 'tiger',
        'HOST': 'localhost'
    }
}
```

5. WORKFLOW IMPLEMENTATION

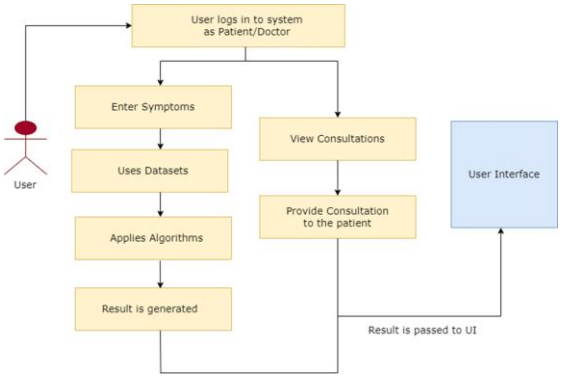


Fig. 5.1 System implementation Flowchart

6. IMPLEMENTATION AND RESULTS

Performance of Algorithms on Training data:

The system was trained on medical record of 4920 patients prone to 41 diseases which was due to the combination of various symptoms. We have considered 95 symptoms out of 132 symptoms to avoid overfitting.

From these results, we can infer that all the three algorithms work exceptionally well on the dataset. However, Decision Tree is perhaps working a little better when compared to the other four algorithms. The accuracy score of each algorithm after training were:

TABLE III. ACCURACY SCORES

Algorithm used	Accuracy score
Decision Tree	.975122
Random Forest	.971121
Gradient Boosting	.971121
K-Nearest	.971121
Naïve Bayes	.971121

Performance of Algorithms on test data:

After training, the system was tested on 41 new patients records considering 95 symptoms. The accuracy score came out to be 97.11 % and the confusion matrix are given as by:

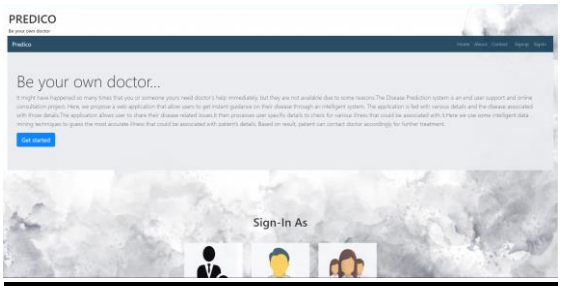
TABLE IV. CONFUSION MATRIX

Confusion Matrix	
Correctly classified	Incorrectly classified
40	1

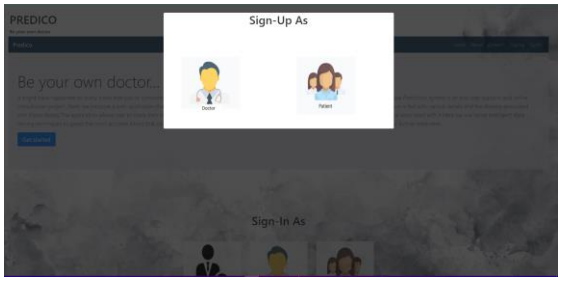
From the above table, we can infer that all the algorithms have equal accuracy score. The accuracy in terms of percentage: 97.11 percentage.

GUI RESULTS:

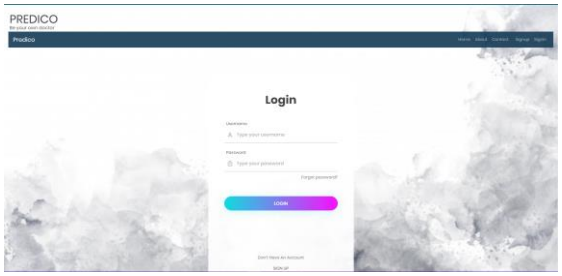
Home Page:



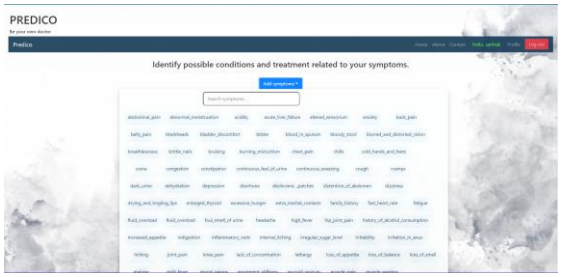
Sign-up Page :



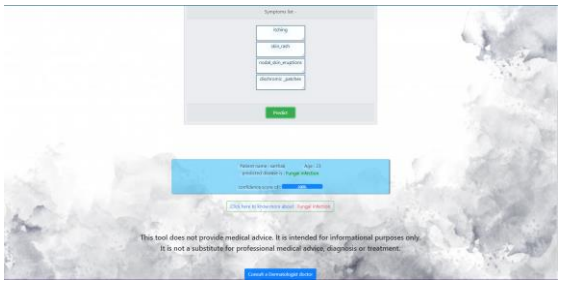
Patient/Doctor Login Page:



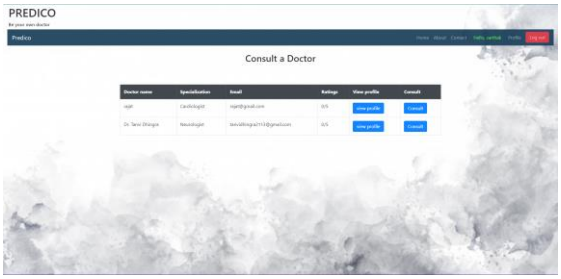
Symptoms Input Page:



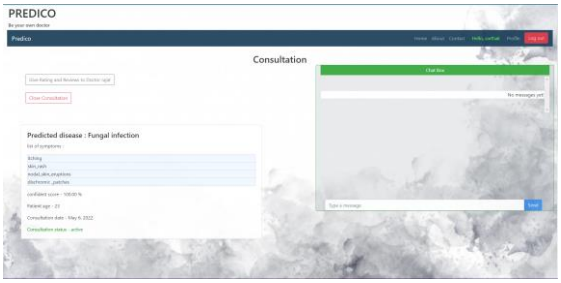
Prediction Result Page



Doctor Consultation Page:



Consultation Chat Page:



8. ACKNOWLEDGEMENT

Our thanks to the Professors, experts and other faculty members who provided useful resources and background to complete this research paper. The Success and outcome of this project were possible by the guidance and support from many people. We are incredibly privileged to have got this all along with the achievement of this paper. It required a lot of effort from each individual involved in this research paper.

9. CONCLUSION AND FUTURE SCOPE

The purpose of this research is to provide a machine learning method to the problem of Disease Detection. This paper reviews the decision tree technique for health prediction. The need for data mining has increased and have very much importance in the field of medicine, as it represents comprehensive process that demands thorough understanding of needs of the healthcare organization. Knowledge gained with the use of decision tree data mining technique can be used to make successful decisions that will improve success of

health care organization and health of the patients. Decision tree technique gives the accurate results for predictions. Health care organizations that use data mining applications have likelihood to predict future requests, needs, desires and conditions of the patients and to form adequate and optimal decisions about the treatments.

Further enhancements that can be done in the system could be converting this web application into an android app. It will be then available to users on mobile basis and its use can be further increased. The feature of getting doctor online on chat so that patients can directly talk to the concerned doctors can enhance the web application. We can upgrade this application by also adding voice and video chat feature. This will make this web application predictable in true sense.

In future work, the creation of more complex ML algorithms is much needed to increase the efficiency of disease prediction. In addition, learning models should be calibrated more often after the training phase for potentially a better performance. Moreover, datasets should be expanded on different demographics to avoid overfitting and increase the accuracy of the deployed models. Finally, more relevant feature selection methods should be used to enhance the performance of the learning models.

10. REFERENCES

[1] Shahadat Uddin , Arif Khan, Md Ekramul Hossain and Mohammad Ali Moni. Comparing different supervised machine learning algorithms for disease prediction. BMC Medical Informatics and Decision Making.

[2] Ahelam Tikotikar and Mallikarjun Kodabagi. A SURVEY ON TECHNIQUE FOR PREDICTION OF DISEASE IN MEDICAL DATA. School of Computing & IT REVA UNIVERSITY

[3] M. Chen, Y. Hao, K. Hwang, L. Wang, and L.Wang,“Disease prediction by machine learning over big data from healthcare communities”, ,” IEEE Access, vol. 5, no. 1, pp. 8869–8879, 2017.

[4] Mr Chintan Shah,Dr. Anjali Jivani, “Comparison Of Data Mining Classification Algorithms for Breast Cancer Prediction”, IEEE-31661

[5] Balasubramanian, Satyabhama, and Balaji Subramanian. "Symptom based disease prediction in medical system by using K-means algorithm." International Journal of Advances in Computer Science and Technology 3.

[6] Pingale, Kedar, et al. "Disease Prediction using Machine Learning." (2019).Mr. Chala Beyene, Prof. Pooja Kamat, “Survey on Prediction and Analysis the Occurrence of Heart Disease Using Data Mining Techniques”, International Journal of Pure and Applied Mathematics, 2018.

[7] Chinmayi Chitnis and Roger Lee. Improving Health-Care Systems by Disease Prediction. 2018 International Conference on Computational Science and Computational Intelligence (CSCI)

[8] A.Davis, D., V.Chawla, N., Blumm, N., Christakis, N., & Barbasi, A. L. (2008). Predicting Individual Disease Risk Based On Medical History. Adam, S., & Parveen, A. (2012). Prediction System For Heart Disease Using Naive Bayes.

[9] P. Groves, B. Kayyali, D. Knott, and S. V. Kuiken, “The ’big data’ revolution in healthcare: Accelerating value and innovation,” 2016.

[10] Data mining, Margaret Rouse, Search SQL Server last accessed 12.09.2018, the article can be found here. https://searchsqlserver.techtarget.com/definition/data- mining

[11] https://www.kaggle.com/neelima98/disease-prediction-using-machine-learning

[12] Supervised and Unsupervised Machine Learning Algorithms, Jason Brownlee, 16.03.2016, published in Machine Learning Algorithms, last accessed 12.09.2018.

[13] Decision trees, scikit-learn.org last accessed 12.09.2018.

[14] RandomForestClassifier, scikit-learn.org last accessed 12.09.2018.

[15] GradientBoostingClassifier, scikit-learn.org last accessed 12.09.2018.

.