IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 09 Issue 07, July 2022
ISSN (Online) 2348 – 7968 | Impact Factor – 6.72
www.ijiset.com

Implementation of Disease Prediction Web Application using the concepts of Machine Learning

Ayushi Arora[1], Swapnil Anil Damate[2], Sai Kishore HR[3], Debee Prasad Rath[4] and Nandana Sreeraj[5]

[1] Full Stack Web Development Researcher, LearnByResearch, Pune, India

[2] Full Stack Web Development Researcher, LearnByResearch, Pune, India

[3] Full Stack Web Development Researcher, LearnByResearch, Banglore, India

[4] Full Stack Web Development Researcher, LearnByResearch, Rayagada, India

[5] Full Stack Web Development Researcher, LearnByResearch, Thiruvananthapuram, India

Abstract

As the world deals with the deadly Corona-virus, remote diagnosis systems are becoming popular and convenient. These systems facilitate enormous advantages like cost-effectiveness, fast and reliable decision support for medical treatment and diagnostics, and prevention from different fatal diseases. These days' healthcare institutions use technology that coincides with the change in environments, aiming to widen the possibility for older and disadvantaged people to access appropriate healthcare services and thus improve their health status and clinical outcome.

An electronic medical record (EMR) is system-enabled health-related information of individuals. EMRs are widely used in healthcare organizations and have many advantages over traditional paper-based medical records, such as their efficiency, reduced storage needs, and ease of computation and diagnosis. In this system, the most challenging things are data security and the accuracy of the disease prediction. The physicians and engineers can use those records and data to create some systems that are beneficial for clinical studies and as well as the patients. It allows the user to keep track of their symptoms over time. The use of symptom trackers also invites a new level of self-management.

This paper proposes an ML model that can predict diseases based on the symptoms. The symptoms entered by the user are extracted and tokenized using the NLTK algorithm. Then those texts are fed to the machine learning model, which will perform the computational work and give the confidence level of the various diseases based on the symptoms. The model provides a probability of the predicted disease, based on which patients can be given suggestions about consultations and medications. The users can also monitor their symptoms change over time, perhaps learn to avoid definite triggers, and ultimately develop the ideal patient-specific treatment with their healthcare professional.

Keywords: **Machine Learning, Random Forest, KNN, XGBoost, NLTK, Recall, F1-Score, Precision.**

1. Introduction

With the world dealing with a life-threatening pandemic, people are in dire need of medical assistance. Medical practitioners are overworked and are unable to diagnose patients having common ailments. There are a number of diseases that are characterized by distinct symptoms, and it becomes easy for the doctors to have a helping hand in taking note of these symptoms and narrowing down the list of diseases associated with the particular symptoms.

Being a part of a developing country, E-facilities in medicine play a vital role in our life. Due to the Covid-19 pandemic, people are afraid to step out and thus rely on the internet and virtual environment to clear their health-related queries. After getting tested for Covid-19 and ruling out one of the possible diseases showing similar symptoms, we tend to dig deep, keeping our health the utmost priority.

Considering today's situation, we propose a web-based ML-backed application that takes a set of symptoms from the patient and then outputs a list of diseases with their probabilities attached. We use NLTK to tokenize the symptoms, extracting a feature vector. This feature vector is passed on to the ML model. Based on this feature vector the most probable disease is extracted and the results are displayed to the patient.

2. Previous Work: The existing Models

In the past few years, there has been significant progress in the field of medicine and diagnosis digitally, with machine learning and neural networks technology. The primary works were noticed in different universities all over the world. Some of these well-known works are discussed in this section.

Faculty of Computer and Information Science, University of Ljubljana, gave an overview of medical data analysis using Machine Learning. It Comments on historical Approaches like Naive Bayesian, Neural networks, Symbolic Learning and tells about the State Of Art Approaches like Assistant R(Decision Trees), Assistant I(Decision Trees), Naive Bayesian Classifier, Semi Naive Bayesian Classifier, Back Propagation with Weight Elimination and k-NN. The best performance is given by Naive Bayesian, Semi Naive Bayesian, and k-NN. [1]

Yuan Luo et al. came up with methods to extract clinical laboratory data from patient testing and apply several machine-learning algorithms to predict ferritin test results using the results from other tests. The methods used include imputation, classification, regression, and univariate analysis. Regression performs best among the lot. [2]

Pannaporn Ketpupong and Krerk Piromposa designed a system that would take a series of symptoms with their corresponding ICD-10-CM code for training, and using NLTK and NLP gave a probability score for each of the diseases listed within the database. They obtained an accuracy of 97.13% with Decision Tree and a True positive rate (TPR) of 89.03% with Neural Network. [3]

Shivam Kalra et al. came up with a method of reading PDF files using OCR and then using the text to predict the various cancer diseases using Term Frequency-Inverse Document Frequency. ICD-10 code was used to train the classifier. XGBoost gave the highest Micro and Macro F-scores, 0.92 and 0.31 respectively. [4]

Hooman H et al. gave a basic knowledge of machine learning categories (supervised, unsupervised, and reinforcement learning) and discussed approaches to the supervised machine learning design along with an overview of common supervised machine learning algorithms. Various Algorithms used were linear regression, logistic regression, Naive Bayes, k-nearest neighbor, support vector machine, random forest, and convolutional neural networks. The conclusion was to develop a Machine learning model for pathology, Identify which algorithm suits them best, then start the development of the model and reiterate through the same. [5]

Y. Deepthi et al. proposed a system that would take data as a symptom with their corresponding prognosis for training and using Naive Bayes, random forests, and decision trees provide the probability score of predicted diseases, that is the prognosis mentioned within the dataset. They obtained 94.6% accuracy with random forests, 84.5% accuracy with naive Bayes, and 78.5% with decision trees. [6]

Md. Martuza Ahmad et al. proposed a system that accumulated raw data that includes basic information, symptoms, and prior diseases for training their model, and the algorithms included are Random Forest, Decision Tree, SVM, and XGBoost and the evaluated metrics are recall value, F1 score, precision, and AUC. The model yields 93% accuracy with SVM, 90% with XGBoost, 85% with a Decision tree, and 80% with Random Forest. [7]

Hiba Hussain et al. used the NLP concept, a synset present in NLTK to get the input of symptoms from the user. The model uses the concepts of Supervised learning in which the KNN and the Decision tree algorithms are applied to the training set. The best of the two, in terms of model's confidence and dataset accuracy was applied on the test set to get the accurate result. The Decision tree and KNN algorithms showed an accuracy of 92.6% and 95.74% respectively. [8]

Sobia Nasir Laique et al. used the OCR/NLP hybrid approach to collect different variables from the EHRs and scanned colonoscopy reports of colorectal cancer patients. The Specificity, Sensitivity, PPV, and accuracy of the model, and extracting required variables were about 98% and above for each variable extracted. [9]

Miao Cui and David Y. Zhang proposed a model which deals with healthcare data that includes the basic information, Whole Slide Image, and other information through the electronic health record system, and all these are integrated with big data for easy analysis and storage. Their main objective is to classify the image into different categories based on their data and information. The algorithms included are SVM, CNN, Random forest. Among these SVM performed well as compared to others with accuracy 93% and AUC 0.98. [10]

3. Methodology

3.1 Web application with integrated ML model

The Web Application developed for data acquisition acquires a set of symptoms from the patients and predicts the most probable disease based on them. The Front end Web Application is designed using the concepts of HTML, CSS, and JavaScript, whereas the back end uses the Python algorithms for performing the tasks. When the patient starts entering the symptoms, the symptoms are displayed in a drop down format as suggestions, and the user can select them as per their requirement. They can enter any number of symptoms to get the appropriate predicted disease.

The Web application uses a data set that comprises 83 symptoms and 49 different diseases. In the first phase, multiple common diseases are predicted using the classifier, along with their probabilities. The input symptoms shared between the user and the Application are processed and sent to the server, where the Machine Learning Model processes it to give the predicted diseases. Fig.1 depicts the flowchart of the Web Application.

In the second phase, the highly trained model will refine the user inputs and deliver the best prognosis to the backend parameters. The server sends the processed response to the user with the help of structured languages, and the predicted diseases are provided to the user. Our model uses the Random Forest Algorithm to train the dataset and test its accuracy before predicting the disease, based on the symptoms extracted from the Web Application.

3.2 Disease Prediction Algorithm

Initially, the algorithm takes in a set of symptoms as the input from the user converting it to an array readable by the model. We used Random Forest Regression Classifier for a better prediction of disease. We divide the dataset into Training dataset and Validation Dataset in the ratio 9:1. The preprocessing is done on the Training and Validation dataset to make it suitable for the model.

For a prediction model in general, classification is considered the most common problem. To solve this issue, we have used a variety of Classification Algorithms, like Random Forests, XGBoost, SVM, etc., and compared the accuracy of each algorithm over the same dataset individually. On testing the accuracy of these models separately, we observed that the SGDC and the KNN classifier gave the highest accuracy. The results of which will be explained in the results section.

In the first process, the model creates a single feature vector from a string of symptoms, provided by the user. It assigns 1 to the input symptoms present in the dataset array and zeros to the remaining. In the second process, the symptoms present in the feature vector are compared with the actual dataset, and the probability of the most common disease is predicted as the output. The flow of the Disease Prediction algorithm is shown in Fig.1.

3.3 System Architecture

The designed Web application takes in the symptoms as input from the user and predicts the most probable disease based on those symptoms. The flow is carried out in different stages, which include a combination of the frontend and the backend application. Fig.1 shows the architecture and flow process of the designed system.

4. Results and Discussions

4.1 Dataset

The Datasets have been downloaded from Kaggle and UCI Repository and the changes have been made manually by the researchers based on the model requirement. Our Dataset consists of about 83 symptoms and 49 different diseases based on the prognosis of these symptoms.

4.2 Results

The results obtained from trying out four different algorithms are present in this section. The algorithms used are:

- **Random Forest Classifier:** Random forests are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest.
- **K-Nearest Neighbors:** K Nearest Neighbors classification algorithm is a supervised machine learning algorithm which classifies a sample based on its similarity with other data points. Similarity is calculated using the distance formula. There are many distance based formulas like Manhattan Distance, Minkowski Distance etc.
- **XGBoost Classifier:** XGBoost Classifier is an ensemble learning method which combines the result of many models choosing to not trust one model. The performance of XGBoost on structured tabular data is unprecedented. The unique features of XGBoost like Regularization, which penalizes bad models, and Handling Sparse Data are useful to our approach.

From Table 1, we observe that KNN is giving us the highest result on the validation dataset. XGBoost. Random Forest and SGDC are performing similarly indicating that the ensemble of trees in XGBoost are giving more or less the same results.

4.3 Metrics for Performance Measurement

- **F1-Score:** F1 score is the weighted average of Precision and Recall. It takes both False Negatives and False Positives into account making it more robust than Accuracy, Precision and Recall.
- **Recall:** Recall is the ratio of correctly predicted positive observations to the all observations in actual Positive class.
- **Precision:** Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.
- **F1 Score Micro:** Micro averaging of F1 score is a common variation of F1 Score when there are more than 2 classes in the dataset. It tries to concentrate on the more common classes and gives appropriate weights with respect to class occurrences. F1 score micro works by averaging the F1 score over all the classes.

  The relationship between the Metrics of Performance Measurement is shown using Equation (1).

5. Tables, Figures and Equations

5.1 Tables and Figures

Algorithms	Random Forest	KNN	SGDC	XGBoost
F1-Score (micro)	0.77	0.97	0.87	0.71

Table 1: Results

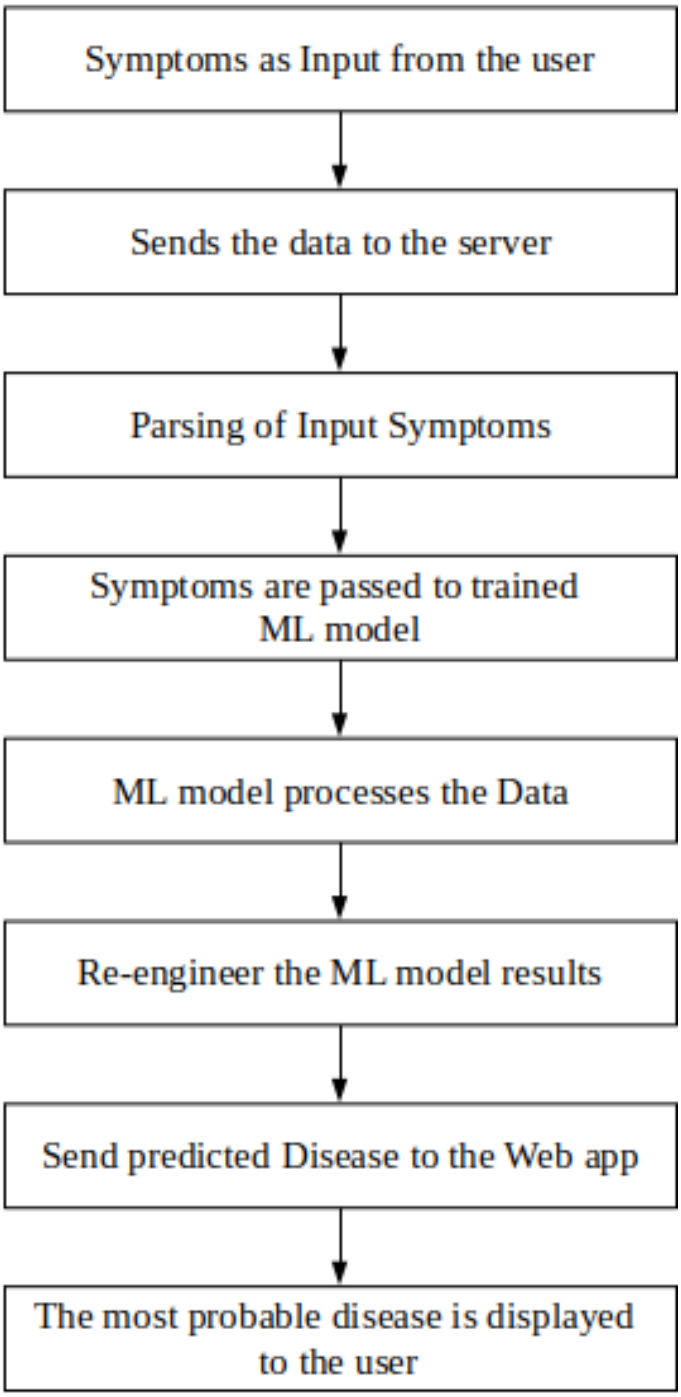

Figure 1: Flowchart of the system

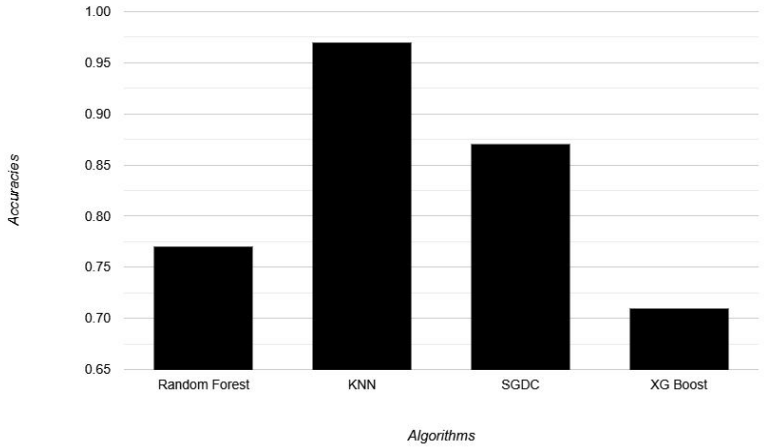

Figure 2: Accuracies

5.2 Equations

F1-Score = 2 * (Precision * Recall) / (Precision + Recall)

6. Conclusions

This study shows that quick prediction of diseases from their symptoms is vital to take necessary precautions to protect ourselves from different diseases. In this study, we used various machine learning algorithms like XGBoost, Random forest classifier, SVM, KNN classifier for the predictive analysis of the subject, but among these, KNN classifiers gave the highest accuracy when compared to other classification algorithms. The web application inputs a list of symptoms through a user-friendly interface and suggests the diseases with the highest probability. The web application is designed based on the concepts of HTML, CSS, and JavaScript, which accounts for taking a set of symptoms from the user as input.

In this paper, to drive the research work, we have designed our dataset by taking inspiration from a few datasets from some of the data repositories. In our dataset, we considered 83 symptoms mapped to 49 diseases based on the prognosis of these symptoms.

The future scope of the research extends to taking scanned images of pathology reports, from which the ML model can consider different parameters and thereby give a diagnosis. But the challenge, in this case, is the preparation of the dataset with the pathology reports because it is tough to get the scanned pathology reports from any data repositories.

References

[1] I. Kononenko, “Machine learning for medical diagnosis: History, state of the art and perspective,” Artif. Intell. Med., vol. 23, no. 1, pp. 89–109, 2001, doi: 10.1016/S0933-3657(01)00077-X.

[2] Y. Luo, P. Szolovits, A. S. Dighe, and J. M. Baron, “Using machine learning to predict laboratory test results,” Am. J. Clin. Pathol., vol. 145, no. 6, pp. 778–788, 2016, doi: 10.1093/ajcp/aqw064.

[3] P. Ketpupong and K. Piromsopa, “Applying Text Mining for Classifying Disease from Symptoms,” Isc. 2018 - 18th Int. Symp. Commun. Inf. Technol., no. Iscit, pp. 467–472, 2018, doi: 10.1109/ISCIT.2018.8587993.

[4] S. Kalra, L. Li, and H. R. Tizhoosh, “Automatic Classification of Pathology Reports using TF-IDF Features,” pp. 1–4, 2019, [Online]. Available: http://arxiv.org/abs/1903..

[5] H. H. Rashidi, N. K. Tran, E. V. Betts, L. P. Howell, and R. Green, "Artificial Intelligence and Machine Learning in Pathology: The Present Landscape of Supervised Methods," Acad. Pathol., vol. 6, 2019, doi: 10.1177/2374289519873088.

[6] Y. Deepthi, K. P. Kalyan, M. Vyas, K. Radhika, D. K. Babu, and N. V. Krishna Rao, "Disease prediction based on symptoms using machine learning," Lect. Notes Electr. Eng., vol. 664, pp. 561–569, 2020, doi: 10.1007/978-981-15-5089-8_55.

[7] M. Ahamad, S. Aktar, S. Uddin, and P. Liò, "Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID- 19 . The COVID-19 resource centre is hosted on Elsevier Connect , the company ' s public news and information ," no. January, 2020.

[8] H. Hussain, K. Aswani, M. Gupta, and G. T. Thampi, "Implementation of Disease Prediction Chatbot and Report Analyzer using the Concepts of NLP , Machine Learning and OCR," pp. 1814–1819, 2020.

[9] S. N. Laique et al., "Application of optical character recognition with natural language processing for large-scale quality metric data extraction in colonoscopy reports," Gastrointest. Endosc., vol. 93, no. 3, pp. 750–757, 2021, doi: 10.1016/j.gie.2020.08.038.

[10] M. Cui and D. Y. Zhang, "Artificial intelligence and computational pathology," Lab. Investig., vol. 101, no. 4, pp. 412–422, 2021, doi: 10.1038/s41374-020-00514-0.