iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** VII **Month of publication:** July 2023

DOI: https://doi.org/10.22214/ijraset.2023.50550

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VII Jul 2023- Available at www.ijraset.com

Designing of Multiple Disease Prediction Model by using Machine Learning and Spyder API

K. Himansu[1], S. Suresh Kumar[2], K. SuryaKala[3], J. Sai Mohan[4]
Department of ECE, Lendi Institute of Engineering and Technology, Jonnada

Abstract: ***Many of the existing machine learning models for health care analysis are concentrating on one disease per analysis. Like one analysis is for diabetes, one for cancer, one for skin diseases like that. There is no common system where one analysis can perform more than one disease prediction. In case of doctor is not available we can use this model. In this model we are proposing a system which used to predict multiple diseases by using Spyder API.***

In this model we used to analyse Diabetes prediction, Heart disease prediction and parkinson's disease prediction analysis. We also developed a model in extension with that based on symptoms it can predict the diseases. To implement multiple disease analysis used machine learning algorithms, streamlit and Spyder API. Python pickling is used to save the model behaviour and python unpickling is used to load the pickle file whenever required. The importance of this model analysis in while analysing the diseases all the parameters which causes the disease is included so it possible to detect the maximum effects which the disease will cause. For example for diabetes analysis in many existing systems considered few parameters like age, sex, bmi, insulin, glucose, blood pressure and pregnancies are considered.

The importance of this analysis to analyse the maximum diseases, so that to monitor the patient's condition and warn the patients in advance to decrease mortality ratio.

Keywords: pickling, insulin, spyder API, disease diagnosis, glucose ,streamlit.

I. INTRODUCTION

In this digital world, data is an asset, and enormous data was generated in all the fields. Data in the healthcare industry consists of all the information related to patients. Here a general architecture has been proposed for predicting the disease in the healthcare industry. Many of the existing models are concentrating on one disease per analysis. Like one analysis for diabetes analysis, one for cancer analysis, one for skin diseases like that. There is no common system present that can analyze more than one disease at a time. Thus, we are concentrating on providing immediate and accurate disease predictions to the users about the symptoms they enter along with the disease predicted. So, we are proposing a system which used to predict multiple diseases by using Spyder API. In this system, we are going to analyze Diabetes, Heart, and parkinson disease analysis. Later many more diseases can be included. In extension with that we also developed common disease prediction model which predicts the disease based on symptoms To implement multiple disease prediction systems we are going to use machine learning algorithms, and Spyder API. Python pickling is used to save the behavior of the model. The importance of this system analysis is that while analyzing the diseases all the parameters which cause the disease is included so it is possible to detect the disease efficiently and more accurately. The final model's behavior will be saved as a python pickle file.

II. LITERATURE SURVEY

A. Common Diseases

Dahiwade et al. [9] proposed a ML based system that predicts common diseases. The symptoms dataset was imported from the UCI ML depository, where it contained symptoms of many common diseases. The system used CNN and KNN as classification techniques to achieve multiple diseases prediction. Moreover, the proposed solution was supplemented with more information that concerned the living habits of the tested patient, which proved to be helpful in understanding the level of risk attached to the predicted disease. Dahiwade et al. [9] compared the results between KNN and CNN algorithm in terms of processing time and accuracy. The accuracy and processing time of CNN were 84.5% and 11.1 seconds, respectively. The statistics proved that KNN algorithm is under performing compared to CNN algorithm. In light of this study, the findings of Chen et al. [10] also agreed that CNN outperformed typical supervised algorithms such as KNN, NB, and DT. The authors concluded that the proposed model scored higher in terms of accuracy, which is explained by the capability of the model to detect complex nonlinear relationships in the feature space.

©IJRASET: All Rights are Reserved

Moreover, CNN detects features with high importance that renders better description of the disease, which enables it to accurately predict diseases with high complexity [9], [10]. This conclusion is well supported and backed with empirical observations and statistical arguments. Nonetheless, the presented models lacked details, for instance, Neural Networks parameters such as network size, architecture type, learning rate and back propagation algorithm, etc. In addition, the analysis of the performances is only evaluated in terms of accuracy, which debunks the validity of the presented findings [9]. Moreover, the authors did not take into consideration the bias problem that is faced by the tested algorithms [9], [10]. In illustration, the incorporation of more feature variables could immensely ameliorate the performance metrics of under performed algorithms [11].

B. Heart Diseases

Marimuthu et al. [16] aimed to predict heart diseases using supervised ML techniques. The authors structured the attributes of data as gender, age, chest pain, gender, target and slope [16]. The applied ML algorithms that were deployed are DT, KNN, LR and NB. As per analysis, the LR algorithm gave a high accuracy of 86.89%, which deemed to be the most effective compared to the other mentioned algorithms. In 2018, Dwivedi [17] attempted to add more precision to the prediction of heart diseases by accounting for additional parameters such as Resting blood pressure, Serum Cholesterol in mg/dl, and Maximum Heart Rate achieved. The used dataset was imported from the UCI ML laboratory; it was comprised with 120 samples that were heart disease positive, and 150 samples that were heart disease negative. Dwivedi attempted to evaluate the performance of Artificial Neural Networks (ANN), SVM, KNN, NB, LR and Classification Tree. At the appliance of tenfold cross validation, the results showed that LR has the highest classification accuracy and sensitivity, which shows high dependability at detecting heart diseases [17]. This conclusion is strengthened by the findings of Polaraju [18] and Vahid et al. [19], where the Logistic Regression outperformed other techniques such as ANN, SVM, and Adaboost. The studies excelled in conducting an extensive analysis on the ML models. For instance, various hyper-parameters were tested at each ML algorithm to converge to the best possible accuracy and precision values. Despite that advantage, the small size of the imported datasets constraints the learning models from targeting diseases with higher accuracy and precision.

C. Parkinson's Disease

Chen et al. [22] presented an effective diagnosis system using Fuzzy k-Nearest Neighbor (FKNN) for the diagnosis of Parkinson's disease (PD) . The study focused on comparing the proposed SVM-based and the FKNN-based approaches. the Principal Component Analysis (PCA) was utilized to assemble the most discriminated features for the construction of an optimal FKNN model. The dataset was taken from the UCI depository, and it recorded numerous biomedical voice measurement ranging from 31 people, 24 with PD. The experimental findings have indicated that the FKNN approach advantageously achieves over the SVM methodology in terms of sensitivity, accuracy, and specificity. In line of this study, Behroozi [23] aimed to propose a new classification framework to diagnose PD, which was enhanced by a filter-based feature selection algorithm that increased the classification accuracy up to 15%. The classification of the framework was characterized by applying independent classifiers for each subset of the dataset to account for the loss of valuable information. The chosen classifiers were KNN, SVM, Discriminant Analysis and NB. The results showed that SVM achieved the highest in all the performance metrics. In addition, Eskidere [24] concentrated on tracking the progression of PD by discussing the performance of SVM with other classifiers such as Least Square Support Vector (LS-SVM), General Regression Neural Network (GRNN) and Multi-layer Perceptron Neural Network (MLPNN). The findings indicated that LS-SVM is the highest performing model. This conclusion is strengthened by the adequate comparison of decoders with their optimal performance metric [25]. According to Lavesson [25], various ML algorithms are designed to optimize numerous performance metrics (e.g., Neural Networks optimizes squared error whereas KNN and SVM optimize accuracy). Furthermore, the authors are particularly good at proposing frameworks with details. For example, SVMs parameters such as the kernel.

III. PROBLEM SYSTEM

Many of the existing machine learning models for health care analysis are concentrating on one disease per analysis. For example first is for liver analysis, one for cancer analysis, one for lung diseases like that. If a user wants to predict more than one disease, he/she has to go through different sites. There is no common system where one analysis can perform more than one disease prediction. Some of the models have lower accuracy which can seriously affect patient's health. When an organization wants to analyse their patient's health reports, they have to deploy many models which in turn increases the cost as well as time Some of the existing systems consider very few parameters which can yield false results.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VII Jul 2023- Available at www.ijraset.com

IV. PROPOSED SYSTEM

In multiple disease prediction, it is possible to predict more than one disease at a time. So the user doesn't need to traverse different sites in order to predict the diseases. We are taking three diseases that are Parkinson, Diabetes, and Heart . As all the three diseases are correlated to each other. To implement multiple disease analyses we are going to use machine learning algorithms. When the user is accessing this API, the user has to send the parameters of the disease

Block Diagram

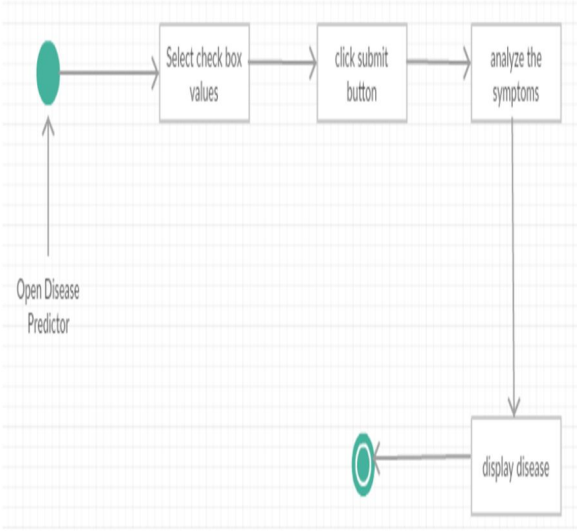


Fig 1:Block diagram of disease predictor

Sequence Diagram

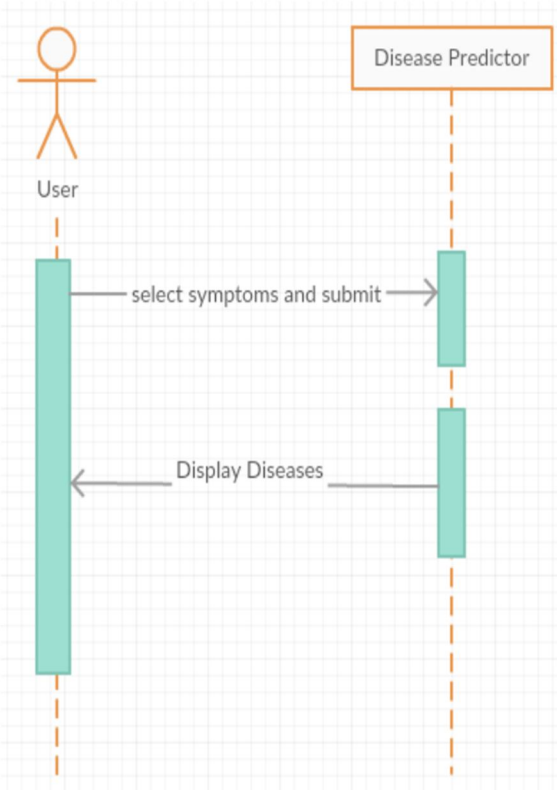


Fig 2:Sequence diagram

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

V. SYSTEM ANALYSIS

A. *Functional Requirement*

1) The system allows the patient to predict the disease.

2) The user adds the input for the particular disease and based on the trained model of the user input the output will be displayed .

B. *Non Functional Requirement*

1) The website will provide range of the values during the prediction of the disease.

2) The website should be reliable and consistent.

VI. IMPLEMENTATION

A. *Algorithm*

1) KNN Algorithm

The working of the K-NN algorithm is as followed:

- Step-1: Start to select the K value for example k=5
- Step-2: Then we will find the Euclidean distance between the points. It is calculated by the as:

$$\textbf{\textit{Euclidiean distance}}: \sqrt{[\mathbf{x_2} - \mathbf{x_1}]^2 + [\mathbf{y_2} - \mathbf{y_1}]^2}$$

- Step-3: Then we will calculate the Euclidean distance of the nearest neighbour.
- Step-4: Then count the number of the data points in each category .For example found three values for Category A and two values for category B.
- Step-5: Then assign the new point to the category having maximum number of neighbours. For example Category A has highest number of neighbour so we will assign the new data point to category A.
- Step-6: So finally our Knn model is ready.

2) SVM Algorithm

- Step-1: Import relevant libraries.
- Step-2:Read in data, perform
  Exploratory Data Analysis (EDA)

- Step-3: Create feature (X) and target (y) dataset.
- Step-4:Split data to 80:20 ratio,and perform model selection.

- Step-5:Optimised model is ready.

3) Logistic Regression Algorithm:

The working of Logistic Regression is as followed:

- Step-1:Importing required libraries.
- Step-2:Data preparation.
- Step-3:Dealing with the missing values.
- Step-4:Exploratory visual analysis.
- Step-5:Modelling the data.
- Step-6:Intepreting:odds ratio,confidence.
- Step-7:Splitting data: Train and Test.
- Step-8:Model evaluation,
- Step-9:so finally it predicts the probability of occurrence of a disease.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VII Jul 2023- Available at www.ijraset.com

VII. RESULT

A. *Diabetes Disease*

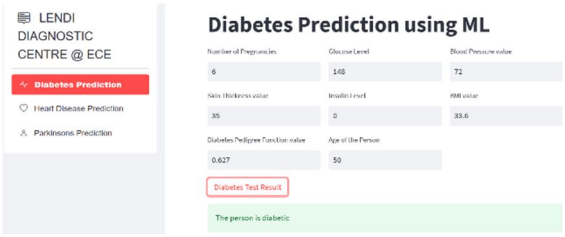


Fig 3:Result prediction of diabetes

B. *Heart Disease*

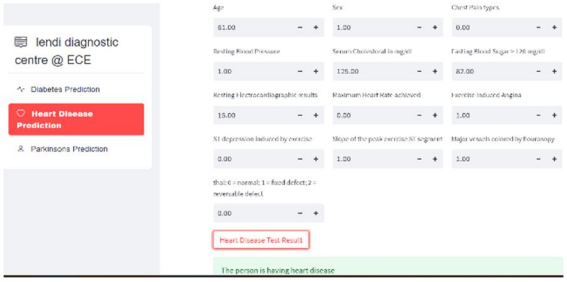


Fig 4:Result prediction of heart disease

C. *Parkinson Disease*

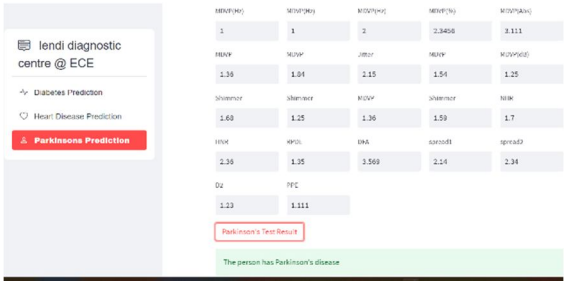


Fig 5:result prediction of parkinson disease

D. *Disease Prediction Based on Symptoms*

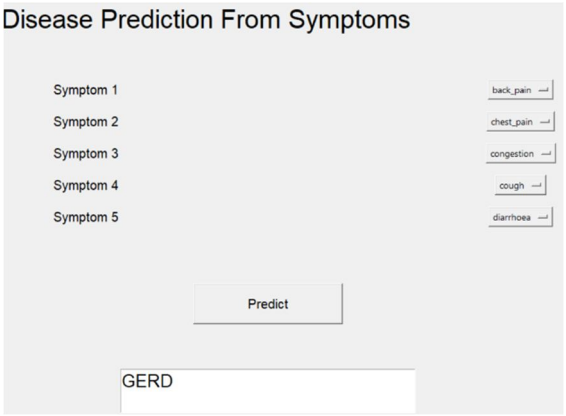


Fig 6: Result prediction of common diseases based on symptoms

©IJRASET: All Rights are Reserved

E. Applications

1) In multiple diseases prediction system a user can analyse more than one disease on a single website.

2) The user doesn't need to traverse different places in order to predict whether he/she has a particular disease or not.

VIII. RESULTS IN JUPYTER

Making a Predictive System

```
In [32]: input_data = (5,166,72,19,175,25.8,0.587,51)

         # changing the input_data to numpy array
         input_data_as_numpy_array = np.asarray(input_data)

         # reshape the array as we are predicting for one instance
         input_data_reshaped = input_data_as_numpy_array.reshape(1,-1)

         # standardize the input data
         std_data = scaler.transform(input_data_reshaped)
         print(std_data)

         prediction = classifier.predict(std_data)
         print(prediction)

         if (prediction[0] == 0):
           print('The person is not diabetic')
         else:
           print('The person is diabetic')

         [[ 0.3429808   1.41167241  0.14964075 -0.09637905  0.82661621 -0.78595734
            0.34768723  1.51108316]]
         [1]
         The person is diabetic
```

Fig 7:Result prediction of diabetes in jupyter

```
In [18]: print('Accuracy on Training data : ', training_data_accuracy)

         Accuracy on Training data :  0.8512396694214877

In [24]: # accuracy on test data
         X_test_prediction = model.predict(X_test)
         test_data_accuracy = accuracy_score(X_test_prediction, Y_test)

In [25]: print('Accuracy on Test data : ', test_data_accuracy)

         Accuracy on Test data :  0.819672131147541

In [31]: input_data = (63,1,3,145,233,1,0,150,0,2.3,0,0,1)

         # change the input data to a numpy array
         input_data_as_numpy_array= np.asarray(input_data)

         # reshape the numpy array as we are predicting for only on instance
         input_data_reshaped = input_data_as_numpy_array.reshape(1,-1)

         prediction = model.predict(input_data_reshaped)
         print(prediction)

         if (prediction[0]== 0):
           print('The Person does not have a Heart Disease')
         else:
           print('The Person has Heart Disease')

         [1]
         The Person has Heart Disease
```

Fig 8:Result prediction of heart disease in jupyter

```
In [24]: print('Accuracy score of test data : ', test_data_accuracy)

         Accuracy score of test data :  0.8717948717948718

In [26]: input_data = (197.07600,206.89600,192.05500,0.00289,0.00001,0.00166,0.00168,0.00498,0.01098,0.09700,0.00563,0.00680,0.00802,0.016

         # changing input data to a numpy array
         input_data_as_numpy_array = np.asarray(input_data)

         # reshape the numpy array
         input_data_reshaped = input_data_as_numpy_array.reshape(1,-1)

         # standardize the data
         std_data = scaler.transform(input_data_reshaped)

         prediction = model.predict(std_data)
         print(prediction)


         if (prediction[0] == 1):
           print("The Person does not have Parkinsons Disease")

         else:
           print("The Person has Parkinsons Disease")

         [0]
         The Person has Parkinsons Disease
```

Fig 9:Result prediction of parkinsons disease in jupyter

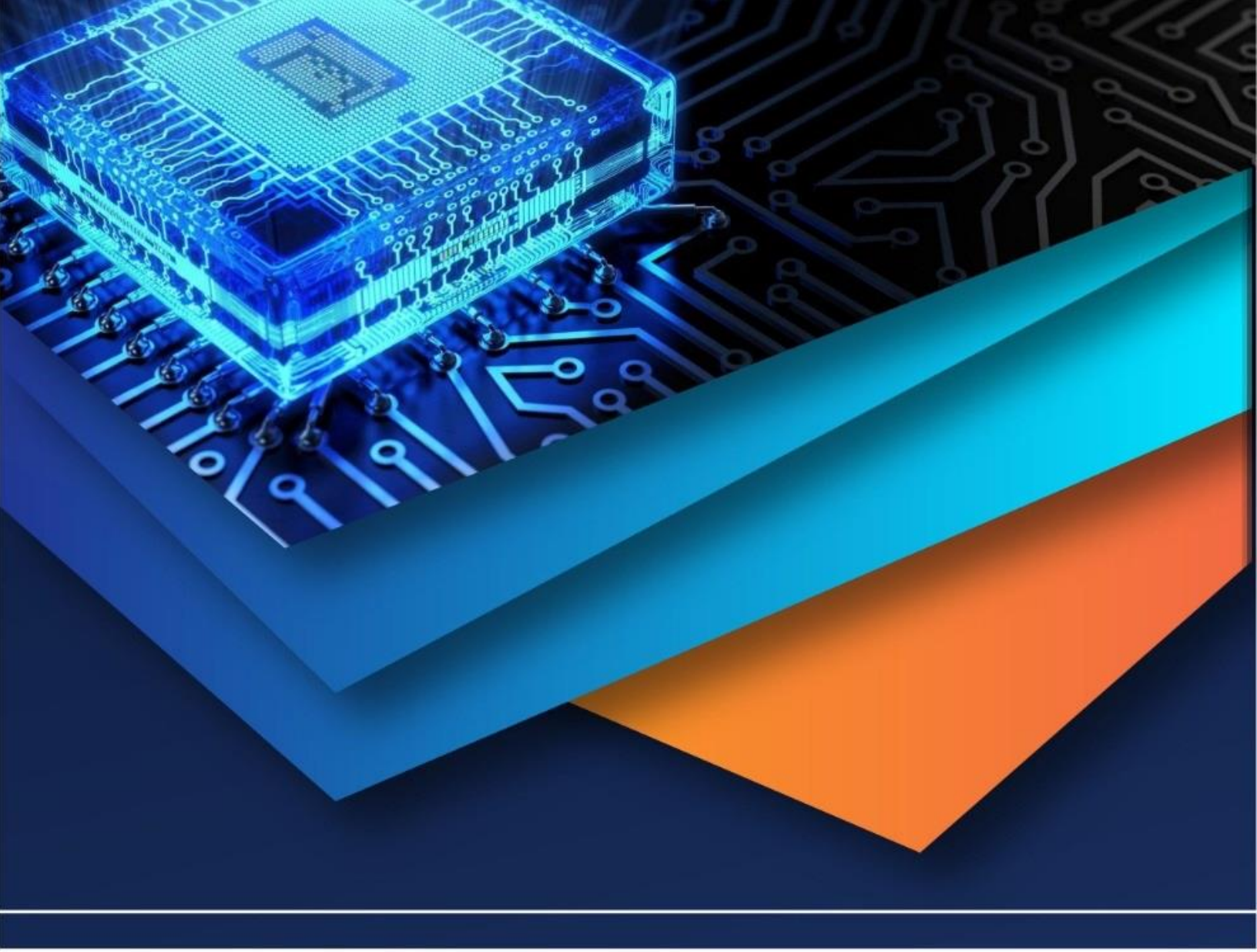
IX. CONCLUSION

The main objective of this project was to create a system that would predict more than one disease and do so with high accuracy. Because of this project the user doesn't need to traverse different websites which saves time as well. Diseases if predicted early can increase your life expectancy as well as save you from financial troubles. For this purpose, we have used various machine learning algorithms like Logistic Regression, SVM, and K nearest neighbor (KNN) to achieve maximum accuracy.

REFERENCES

[1] Gavhane, G. Kokkula, I. Pandya, and K. Devadkar, "Prediction of heart disease using machine learning," in 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2018, pp. 1275–1278.
[2] Y. Hasija, N. Garg, and S. Sourav, "Automated detection of dermatological disorders through image-processing and machine learning," in 2017 International Conference on Intelligent Sustainable Systems (ICISS), 2017, pp. 1047–1051.
[3] S. Uddin, A. Khan, M. E. Hossain, and M. A. Moni, "Comparing different supervised machine learning algorithms for disease prediction," BMC Medical Informatics and Decision Making, vol. 19, no. 1, pp. 1– 16, 2019.
[4] R. Katarya and P. Srinivas, "Predicting heart disease at early stages using machine learning: A survey," in 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), 2020, pp. 302–305.
[5] P. S. Kohli and S. Arora, "Application of machine learning in disease prediction," in 2018 4th International Conference on Computing Communication and Automation (ICCCA), 2018, pp. 1–4.
[6] M. Patil, V. B. Lobo, P. Puranik, A. Pawaskar, A. Pai, and R. Mishra, "A proposed model for lifestyle disease prediction using support vector machine," in 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2018, pp. 1–6.
[7] F. Q. Yuan, "Critical issues of applying machine learning to condition monitoring for failure diagnosis," in 2016 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), 2016, pp. 1903–1907.
[8] S. Ismaeel, A. Miri, and D. Chourishi, "Using the extreme learning machine (elm) technique for heart disease diagnosis," in 2015 IEEE Canada International Humanitarian Technology Conference (IHTC2015), 2015, pp. 1–3.
[9] D. Dahiwade, G. Patle, and E. Meshram, "Designing disease prediction model using machine learning approach," Proceedings of the 3rd International Conference on Computing Methodologies and Communication, ICCMC 2019, no. Iccmc, pp. 1211–1215, 2019.
[10] S. Jadhav, R. Kasar, N. Lade, M. Patil, and S. Kolte, "Disease Prediction by Machine Learning from Healthcare Communities," International Journal of Scientific Research in Science and Technology, pp. 29–35, 2019.
[11] R. Saravanan and P. Sujatha, "A state of art techniques on machine learning algorithms: A perspective of supervised learning approaches in data classification," in 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), 2018, pp. 945– 949.
[12] Y. Amirgaliyev, S. Shamiluulu, and A. Serek, "Analysis of chronic kidney disease dataset by applying machine learning methods," in 2018 IEEE 12th International Conference on Application of Information and Communication Technologies (AICT), 2018, pp. 1–4.
[13] V. S and D. S, "Data Mining Classification Algorithms for Kidney Disease Prediction," International Journal on Cybernetics & Informatics, vol. 4, no. 4, pp. 13–25, 2015.

[14] A. Charleonnan, T. Fufaung, T. Niyomwong, W. Chokchueypattanakit, S. Suwannawach, and N. Ninchawee, "Predictive analytics for chronic kidney disease using machine learning techniques," 2016 Management and Innovation Technology International Conference, MITiCON 2016, pp. MIT80–MIT83, 2017.
[15] P. Kotturu, V. V. Sasank, G. Supriya, C. S. Manoj, and M. V. Maheshwarredy, "Prediction of chronic kidney disease using machine learning techniques," International Journal of Advanced Science and Technology, vol. 28, no. 16, pp. 1436–1443, 2019.
[16] M. Marimuthu, M. Abinaya, K. S., K. Madhankumar, and V. Pavithra, "A Review on Heart Disease Prediction using Machine Learning and Data Analytics Approach," International Journal of Computer Applications, vol. 181, no. 18, pp. 20–25, 2018.
[17] A. K. Dwivedi, "Performance evaluation of different machine learning techniques for prediction of heart disease," Neural Computing and Applications, vol. 29, no. 10, pp. 685–693, 2018.
[18] K. Polaraju, D. Durga Prasad, and M. Tech Scholar, "Prediction of Heart Disease using Multiple Linear Regression Model," International Journal of Engineering Development and Research, vol. 5, no. 4, pp. 2321–9939, 2017. [Online]. Available: www.ijedr.org
[19] S. Pouriyeh, S. Vahid, G. Sannino, G. De Pietro, H. Arabnia, and J. Gutierrez, "A comprehensive investigation and comparison of machine learning techniques in the domain of heart disease," in 2017 IEEE Symposium on Computers and Communications (ISCC), 2017, pp. 204– 207.
[20] P. P. Sengar, M. J. Gaikwad, and A. S. Nagdive, "Comparative study of machine learning algorithms for breast cancer prediction," Proceedings of the 3rd International Conference on Smart Systems and Inventive Technology, ICSSIT 2020, pp. 796–801, 2020.
[21] D. Yao, J. Yang, and X. Zhan, "A novel method for disease prediction: Hybrid of random forest and multivariate adaptive regression splines," Journal of Computers (Finland), vol. 8, no. 1, pp. 170–177, 2013.
[22] H. L. Chen, C. C. Huang, X. G. Yu, X. Xu, X. Sun, G. Wang, and S. J. Wang, "An efficient diagnosis system for detection of Parkinson's disease using fuzzy k-nearest neighbor approach," Expert Systems with Applications, vol. 40, no. 1, pp. 263–271, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2012.07.014
[23] M. Behroozi and A. Sami, "A multiple-classifier framework for Parkinson's disease detection based on various vocal tests," International Journal of Telemedicine and Applications, vol. 2016, 2016.
[24] O. Eskidere, F. Ertas¸, and C. Hanilc¸i, "A comparison of regression ¨ methods for remote tracking of Parkinson's disease progression," Expert Systems with Applications, vol. 39, no. 5, pp. 5523–5528, 2012.
[25] N. Lavesson, Evaluation and Analysis of Supervised Learning Algorithms and Classifiers, 2006.
[26] R. Caruana and A. Niculescu-Mizil, "An Empirical Comparison of Supervised Learning Algorithms Using Different Performance Metrics," Proceedings of the 23rd international conference on Machine Learning, pp. 161–168, 2006. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.60.3232



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)