Multiple disease prediction using Machine Learning Algorithms

Srinidhi B[1], T.P.Manasa[2]

[1]M.Tech, Dept. of Computer Science, Banglore Institute of Technology, Bengaluru, India
[2]Assistant Prof, Dept. of Computer Science, Banglore Institute of Technology, Bengaluru, India

---***---

Abstract – **Many of the machine learning models for health care analysis now in use focus on just one disease at a time. Like one analysis is for diabetes, one for heart, and one for brain tumor. A single analysis cannot forecast more than one disease using a same framework. In this paper, a system that uses the Flask Application programming interface (API) to forecast numerous diseases is proposed. This paper proposed a method to examine diabetes, brain tumor, heart disease, and Alzheimer. Later, more disorders might be included, such as covid detection, fever tests, and many others. TensorFlow, Flask API, and machine learning techniques were utilized to implement numerous disease analyses. The model behavior is saved using Python pickling, and the pickle file is loaded when needed using Python unpickling. The significance of this article analysis is that all the factors that contribute to the sickness are considered while analyzing it, making it possible to identify the disease's full range of potential impacts. For instance, many of the existing systems for analyzing diabetes considered a few parameters such as age, sex, BMI, and insulin. But in our proposed system we have considered additional factors like number of pregnancies, Glucose concentration, skin thickness, heart rate/pulse rate, and diabetes pedigree which makes our model more efficient compared to existing model.**

The final model's behavior will be recorded in a pickle file in Python and the Flask API will be created. The parameters of the disease must be sent together with the disease name when using this API. The relevant model will be called by the Flask API, which then returns the patient's state. The significance of this study is to examine the most diseases possible to keep track of patients' conditions and forewarn them when necessary to reduce the death rate.

Keywords — Machine Learning, Flask API, Convolutional Neural Network, XG-Boost.

I. INTRODUCTION

Healthcare is increasingly adopting the artificial intelligence (AI) technologies that are pervasive in modern business and daily life. Artificial intelligence in healthcare has the potential to aid providers in many areas of patient care and operational procedures enabling them to build on current solutions and solve problems more quickly. The majority of AI and healthcare technologies are highly relevant to the healthcare industry, yet hospitals and other healthcare organizations may employ quite different strategies. Precision medicine is the most widely used use of conventional machine learning in the field of artificial intelligence in healthcare. It is a big step forward for many healthcare organizations to be able to forecast which treatment approaches would be most effective with patients based on their characteristics and the treatment framework.

And even though some articles on the use of artificial intelligence in healthcare suggest that it can perform just as well as or better than humans at certain procedures, like diagnosing disease, it will be a considerable amount of time before AI in healthcare replaces humans for a wide range of medical tasks. Machine learning and precision medicine applications, which make up the majority of AI in healthcare, require data for training with known outcomes. We call this supervised learning.

Many analyses of the health care industry's current systems only took one ailment at a time. For instance, articles are used to analyze diabetes, to analyze brain tumor, and to forecast heart disease. Most articles concentrate on a specific disease. Any institution that wishes to analyze the health reports of its patients must use a variety of models. The method used in the current system is helpful for studying only a certain ailment. Today's mortality has grown as a result of the failure to accurately diagnose diseases. Even after being treated for one disease, a patient may still be afflicted with another. When analyzing the disease, several current methods employed a limited number of parameters. Because of this, it might not be feasible to pinpoint the diseases that will be brought on by the disease's effects. For instance, diabetes increases the risk of developing dementia, hearing loss, retinopathy, neuropathy, and heart disease. The analysis of diabetes, brain tumor, heart disease, and Alzheimer data sets were taken into consideration in this paper. In the future, several other diseases may be added, such as skin conditions, diseases linked to fever, and many more. This adaptable analysis later encompassed several disorders for examination.

The model file associated with the analysis of the new disease must be added by the developer when adding any new disease analysis to this API. The developer must set up Python picking to store model behavior when creating new diseases. The developer can load a pickled file to obtain the

model behavior while utilizing this Flask API. When a user wishes to analyze a patient's health, they may either forecast a specific disease or, if the report contains factors that are used to predict other diseases, this analysis will provide the most appropriate disease identifications. By forewarning patients depending on their health circumstances, this article aims to stop the mortality ratio from rising day by day. The cost of patient analysis can be decreased since there are several diseases models and forecasts made at one location.

The section I explains the Introduction of multiple disease prediction using machine learning. Section II presents the literature review of existing systems and Section III present proposed system architecture and implementation details Section IV presents results and discussion of proposed system. Section V concludes our proposed system. While at the end list of references paper are presented

II. LITRATURE SURVEY

M. Chen suggested [1] a novel Convolutional neural network (CNN) based multimodal disease risk prediction method based on hospital structured and unstructured data. M. Chen, Y. Hao, K. Hwang, L. Wang, and L. Wang developed a disease prediction system for many locations. They predicted the onset of three diseases: diabetes, cerebral infraction, and heart disease. Disease prediction is based on structured data. Prediction of heart disease, diabetes, and cerebral infraction is done using several machine learning algorithms such as naive bayes, decision tree, and K-Nearest Neighbor (KNN) algorithm. The Decision tree method outperforms the Naive Bayes and KNN algorithms. They also indicate whether a patient would suffer from a high risk of cerebral infarction or a low risk of cerebral infarction. They used CNN-based multi model disease risk prediction using text data to estimate the risk of cerebral infraction. The accuracy of CNN-based unimodal disease risk predictions (UDRP) is compared to CNN-based multi model disease risk prediction system. With a quicker speed than a CNN-based unimodal disease risk prediction algorithm, disease prediction accuracy can reach 94.8%. The CNN-based multi model disease risk prediction algorithm processes are identical to those of the CNN-UDRP method, apart from two extra testing phases. This research focuses on both structured and unstructured data datasets. The writer works with unstructured data. Previous work was exclusively based on structured data, and none of the authors dealt with unstructured or semi-structured data. However, this paper is dependent on both structured and unstructured data.

B. Qian, X. Wang, N. Cao, H. Li, and Y.-G. Jiang [2] created an Alzheimer disease risk prediction system using patient electronic health record (EHR) data. They used an active learning setting to address a real-world problem for the patient. This is where the active patient risk model was created. The risk of Alzheimer's disease is used in the active risk prediction method.

Wearable 2.0 is a method presented by IM. Chen, Y. Ma, Y. Li, D. Wu, Y. Zhang, and C. Youn [3] that involves designing washable smart clothing that enhances the quality of experience and quality of service of the next-generation healthcare system. New IoT-based data collecting system was created by Chen. newly developed smart washable cloth with sensors. The doctor used this cloth to record the patient's physiological state. Further analysis is done with the aid of the physiological data. The major components of this washable smart cloth include several sensors, cables, and electrodes. Using these parts, a user can gather data on the patient's physiological state as well as their emotional health status using a cloud-based system. It was able to capture the patient's physiological state with the aid of this fabric. And this data is used for the analysis. They discussed the difficulties encountered when creating the wearable 2.0 architecture. The gathering of physiological data, detrimental psychological impacts, opposition to wireless body area networking, and sustainable massive physiological data collection are among the problems with the current technology. the many activities carried out on files, such as data analysis, monitoring, and prediction. Once more, the author divides the functional elements of wearable 2.0 smart clothing into the following groups: sensors integration, electrical cable-based networking, and digital modules. Numerous applications are covered in this, including the monitoring of chronic diseases, the care of the aged, and the management of emotions.

A cloud-based health Cyber-Physical Systems (CPS) system that maintains the enormous volume of biological data was created by Y. Zhang, M. Qiu, C.-W. Tsai, M. M. Hassan, and A. Alamri [4]. Y. Zhang spoke on the medical industry's rapid rise in data volume. The issue with large data is that information is produced quickly and has a tendency to be stored in many formats. The health-CPS system was created using two technologies, with big data technology coming in second. Data collecting, data administration, and data-oriented layers are the three levels that the system takes into account. The layer that collected the data did it in a certain standard format. the layer of data management utilized for parallel computing and distributed storage. With the aid of the health-cps system, several procedures are carried out by this system. Moreover, this system is aware of various healthcare related services.

A telehealth system was presented by L. Qiu, K. Gai, and M. Qiu in [5], along with a discussion on how to manage a significant volume of hospital data in the cloud. The author of this research recommended improvements to the telehealth system, which are largely focused on the cloud-based data exchange amongst all telehealth providers. However, there are other problems with cloud data sharing, including network bandwidth and virtual machine

switching. For improved data sharing using data sharing ideas, a cloud-based solution to data sharing is recommended in this. Here, the ideal telehealth sharing model has been developed. With the use of this model, the author focuses on temporal restrictions, network capabilities, and transmission probability. For this, the author created a brand-new, ideal method for exchanging massive data. Users are provided with the best method for processing biological data by this algorithm.

The best clinical decision-making approach was proposed by Ajinkya Kunjir, Harshal Sawant, and Nuzhat F. Shaikh [6] and uses patient history to forecast disease. Numerous diseases and an unanticipated pattern of patient state are predicted in this. created a top-notch clinical decision-making tool that is utilized for precise disease prediction based on historical data. The notion of many diseases and an unknown pattern were also determined in it. Pie charts and 2D/3D graphs were utilized in this for visualization purposes.

A comparison of several machine learning techniques, including fuzzy logic, fuzzy neural networks, and decision trees, is provided by S.Leoni Sharmila, C.Dharuman, and P. Venkatesan in their article [7].They use the liver data set to categorize and conduct comparative research. According to a research, Fuzzy Neutral Network performs 91% more accurately than other machine learning algorithms in classifying liver disease datasets. Author is skilled at doing classification extremely well and providing very good performances. Author employed Simplified Fuzzy ARTMAP in a variety of application areas.

The author has concluded that on the supplied data set, machine learning techniques like Naive Bayes and Apriori [8] are very helpful for disease detection. Here, small-volume data, such as symptoms or prior information gleaned from the physical diagnostic, are employed for prediction. This paper's limitations include the inability to consider enormous datasets. Additionally, because medical data is increasing today, it is difficult to identify it.

A CNN- multimodal disease risk predictions MDRP technique was put up by Shraddha Subhash Shirsath [9] to forecast diseases based on a sizable amount of structured and unstructured hospital data. Using a machine learning algorithm (Naive- Bayes), CNN-MDRP focuses on both structured and unstructured data, increasing disease prediction accuracy and speed compared to CNN-UDRP, which only analyses structured data. Big data is considered here.

A new method built on the Artificial neural network (ANN) algorithm was developed for the prediction of heart disease [10]. The researchers developed a categorization-based interactive prediction approach employing an artificial neural network algorithm and accounting for the thirteen most crucial clinical factors. Healthcare professionals can benefit greatly from the recommended strategy, which had an accuracy rate of 80% when it came to predicting heart disease.

To predict heart disease, the fuzzy method with a membership function was applied [11]. The authors sought to remove ambiguity and uncertainty from the data by using the Fuzzy KNN Classifier. The 25 classes and 22 elements in each class were created from the 550-record dataset. The dataset was divided into training and testing halves of equal size. Following the use of pre-processing methods, the fuzzy KNN methodology was applied. Accuracy, precision, and recall were only a few of the assessment variables used to evaluate this method. The results revealed that the fuzzy KNN classifier performed better than the KNN classifier in terms of accuracy.

To assess a big picture collection, Yang et al. [12] presented the CBIR (Content-Based Image Retrieval) approach. This approach looks for a tumor with the same pathological characteristics by using the tumor area as a query. The feature employed by the system was called MID (Margin Information Descriptor). By a tumor's surrounding tissue, it clarifies the image's meaning. Their suggested approach was successful in achieving 89.3% accuracy. However, because their approaches derive a globally linear distance measure, they provide many local predictions for various locations. Additionally, the manual segmentation of the brain tumor was used in the development of their method.

Support vector machines (SVM) with the adaptivity feature were used to identify breast cancer and diabetes [13]. The objective was to provide an automatic, quick, and flexible diagnostic approach utilizing adaptive SVM. The bias value in traditional SVM was modified to provide better results. If-then rules were generated as output from the proposed classifier. Breast cancer and diabetes were both successfully diagnosed using the suggested procedure, which had a 100% accuracy rate for both diagnoses. The development of more effective strategies for modifying the bias value in traditional SVM should be the focus of future research.

III. System Design

A. Proposed System

It is feasible to forecast numerous diseases at once using multiple disease prediction. Therefore, the user does not have to visit several sites to anticipate the ailments. We are focusing on the disorders of the heart, brain tumor, Alzheimer, and diabetes. because there is a correlation between the three disorders. We'll employ machine learning methods and Flask-API to create numerous disease analyses. The user must submit both the disease name and its parameters when requesting access to this API. Flask will call the relevant model and deliver the patient's status.
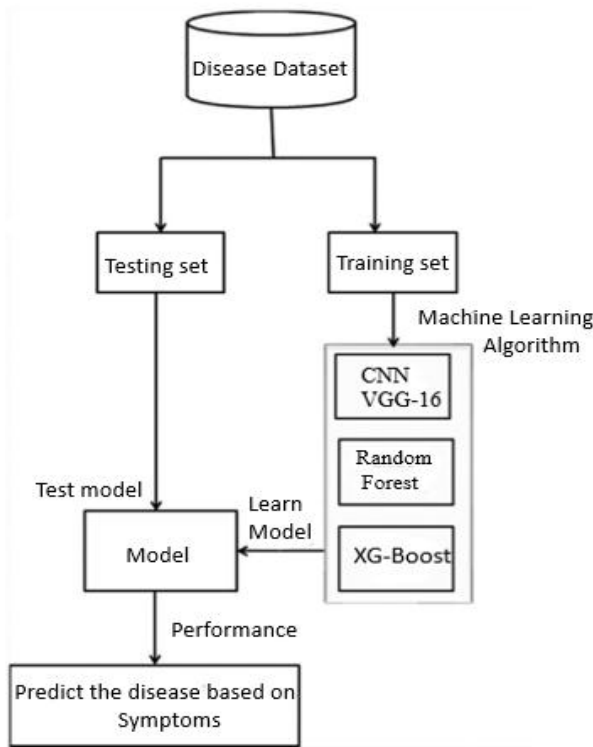
B. Architecture Overview

Fig 3.1; System Architecture

We conducted experiments on four diseases heart, diabetes, brain tumor and Alzheimer. The first step is to import the UCI dataset, PIMA dataset, and Indian liver dataset, respectively, for the datasets for heart disease, diabetes disease, and liver disease. After the dataset has been loaded, each inputted piece of data is shown. After pre-processing the data for visualization, which involves looking for outliers, missing values, and scaling the dataset, the data is divided into training and testing. Next, we used the CNN, XG-Boost, and random forest algorithms on the training dataset before applying what we learned about the classification method to the testing dataset. We will select the algorithm with the highest accuracy for each ailment after applying our expertise. Then, we created a pickle file for each ailment and combined it with the Flask framework to provide the model's output on the website.

C. Algorithms

CNN Algorithm

Step 1: The dataset is first transformed into a vector format.

Step 2: Following that, word embedding was completed, using zero values to fill in the data. Word embedding produces a convolutional layer as its result.

Step 3: We conduct the maximum pooling operation on the convolutional layer after taking it as an input to the pooling layer.

Step 4: Convert the dataset into a fixed-length vector form for Max pooling. The entire connected neural network is coupled to the pooling layer.

Step 5: The classifier, a SoftMax classifier, is linked to the complete connection layer.

XG-Boost Algorithm

The XG-Boost algorithm operates as follows:

Step 1: Creating a single leaf tree is the first step.

Step 2: After computing the average of the target variable as prediction for the first tree, we must determine the residuals using the specified loss function. The residuals for future trees are then derived from the prediction from the first tree.

Step 3: Using the formula to determine the similarity score.

$$\text{Similarity Score} = \frac{\left(\sum \text{Residuals}\right)^2}{N + \lambda}$$

N = No. of Residuals

λ = Regularization Parameter

Step 4: Using the similarity score, we choose the correct node. More homogeneity is seen when the similarity score is greater.

Step 5: The fifth step involves applying the similarity score to the information acquired. Information gain reveals how much homogeneity is obtained by dividing the node at a specific place and helps to distinguish between old and new similarities. The formula used to compute it is:

Information Gain = LS + RS – Similarity for Roots

Where LS= Left Similarity & RS=Right Similarity

Step 6: Using the technique, you may prune and regularize the tree to the appropriate length by adjusting the regularization hyperparameter.

Step 7: Using the Decision Tree you created, we can then forecast the residual values.

Step 8: The learning rate is used to calculate the new set of residuals.

New Residual = Old Residual $+\delta\sum$ Predicted Residual

Step 9: Next, go back to step 1 and carry out step 1 again for each tree.

Random Forest Algorithm

A random forest may be constructed by merging N decision trees, and then it can be used to make predictions for each tree that was produced in the first step.

The random forest operates as follows:

Step-1: First, it will choose K data points at random from the training set.

Step 2: After choosing k data points, create decision trees linked to the chosen data points (Subsets).

Step 3: After that, pick the Nth node for the decision trees you wish to construct.

Step-4: Repeat steps 1 and 2 in step 4.

Step-5: Finding each decision tree's predictions and allocating the fresh data points to the category with the most support is step five.

VGG-16 Algorithm

Step 1: The 16 in VGG16 stands for 16 weighted layers. Thirteen convolutional layers, five Max Pooling layers, three Dense layers, and a total of 21 layers make up VGG16, although only sixteen of them are weight layers, also known as learnable parameters layers.

Step 2: The input tensor size for VGG16 is 224, 244 and has three RGB channels.

Step 3: The most distinctive feature of VGG16 is that it prioritized convolution layers of a 3x3 filter with stride 1 rather than many hyper-parameters and consistently employed the same padding and maxpool layer of a 2x2 filter with stride 2.

Step 4: Throughout the whole design, the convolution and max pool layers are uniformly ordered.

Step 5: There are 64 filters in the Conv-1 Layer, 128 filters in Conv-2, 256 filters in Conv-3, and 512 filters in Conv-4 and Conv-5.

Step 6: A stack of convolutional layers is followed by three Fully Connected (FC) layers, the third of which conducts 1000-way ILSVRC classification and has 1000 channels. The first two FC layers have 4096 channels each (one for each class). The soft-max layer is the last one.

To conclude with, the CNN algorithm, the xg-boost algorithm, ind the random forest algorithm were utilized in the system prediction model since they each provided the highest levels of accuracy. When the patient adds the disease-specific parameter, it will indicate whether the

patient has the ailment in question. The parameters will display the necessary value range, and if the value is outside of that range, is invalid, or is empty, a warning message will appear, advising the user to input a proper value.

IV. RESULTS AND DISCUSSION

A. Experimental Setup

All the experimental cases are developed in Python in a congested environment using Anaconda tools. The competing classification approach and various feature extraction techniques are also used, and the system is configured with an Intel Core i5-6200U processor running at 2.30 GHz and 8GB of RAM.

B. Dataset

The disease dataset are downloaded from Github, kaagle and other ML website. And as per industry standards train set and test are prepared. By using Scikit learn train, test, split method to split the data as 70 % for training and 30 % for testing are divided.

Example of Diabetic Disease:

Diabetes feature train, diabetes feature test,

Diabetes label train, diabetes label test=train test split (diabetes features, diabetes label, test size=0.3, train size=0.7)

C. Evaluation Methods

For the experiment's performance assessment. First, we identify True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). True positive refers to the number of cases successfully forecasted as necessary, false positive refers to the number of instances mistakenly predicted as required, and so on. The following are the four measurements that may be obtained: accuracy, precision, recall, and F1-measure.

Accuracy = (TP+TN)/(TP+TN+FP+FN)

Precision = TP/(TP+FP)

Recall = TP/(TP+FN)

F1 Score = TN/(TN+FP)

D. Result

When the patient adds the disease-specific parameter, it will indicate whether the patient has the ailment in question. The parameters will display the necessary value range, and if the value is outside of that range, is invalid, or is empty, a warning message will appear, advising the user to input a proper value.

When the Flask API is created. The front end can devour the model. It will be confirmed by creating a test web page, here heart disease input screen is shown as an example in Fig 4.1. Once the user hits "Submit Button," whether the patient has heart disease or not will be returned.
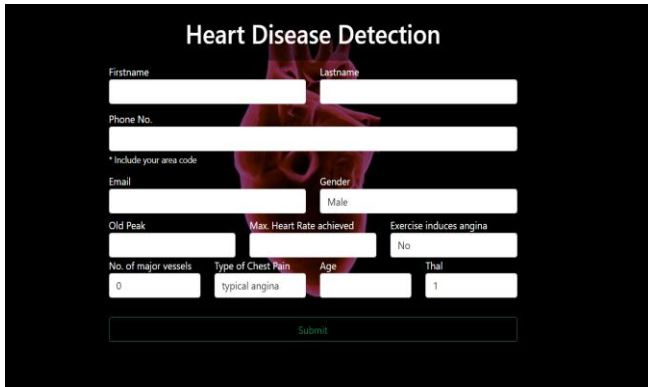

Fig 4.1: Heart Disease Input Screen

The results of the heart disease prediction are shown in Figure 4.2. Once the user has finished filling out the form and clicked "Submit." When the relevant model has been loaded, the user will see the result of heart illness.
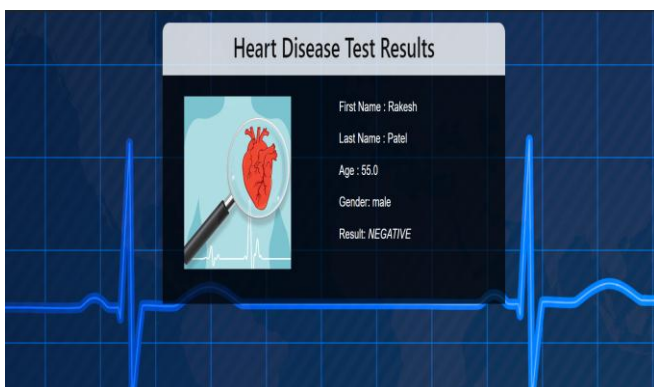
Fig 4.2: Output of patient Status

In this way, Multiple diseases can be predicted simultaneously using a multi-disease prediction model. Here, a disease prediction is made based on user input. The user will be offered the option. A related disease model will be activated and predicted based on the user's given inputs if they wish to forecast a specific disease or if they don't enter any disease types. The benefit of using a multi-disease prediction model in advance is that it can estimate the likelihood that numerous diseases will manifest themselves while also lowering the mortality rate.

V. CONCLUSION

We used standard machine learning algorithms to categorize patient data since, in the modern medical world, medical data is expanding greatly and must be processed to provide precise disease predictions based on symptoms. By providing the input of patient records, which aid in understanding the degree of disease risk prediction, we were able to produce an accurate general disease risk prediction. Because of this technique, disease and risk prediction may be accomplished with little effort and expense. We compare the outcomes of the different algorithms in terms of accuracy and processing time. The accuracy of the CNN algorithm is higher than that of the other machine learning algorithm, and CNN's processing time is lower than that of others. So, in terms of accuracy and timing, CNN is superior to other machine learning algorithm.

In the future, we'll include more diseases and forecast the likelihood that a patient would develop each one.

REFERENCES

[1] M. Chen, Y. Hao, K. Hwang, L. Wang and L. Wang, "Disease Prediction by Machine Learning Over Big Data From Healthcare Communities," in IEEE Access, vol. 5, pp. 8869-8879, 2017, doi: 10.1109/ACCESS.2017.2694446.

[2] Qian, Buyue & Wang, Xiang & Cao, Nan & Li, Hongfei & Jiang, Yu-Gang. (2014). A relative similarity based method for interactive patient risk prediction. Data Mining and Knowledge Discovery. 29. 10.1007/s10618-014-0379-5.

[3] M. Chen, Y. Ma, Y. Li, D. Wu, Y. Zhang and C. -H. Youn, "Wearable 2.0: Enabling Human-Cloud Integration in Next Generation Healthcare Systems," in IEEE Communications Magazine, vol. 55, no. 1, pp. 54-61, January 2017, doi: 10.1109/MCOM.2017.1600410CM.

[4] Y. Zhang, M. Qiu, C. -W. Tsai, M. M. Hassan and A. Alamri, "Health-CPS: Healthcare Cyber-Physical System Assisted by Cloud and Big Data," in IEEE Systems Journal, vol. 11, no. 1, pp. 88-95, March 2017, doi: 10.1109/JSYST.2015.2460747.

[5] Gai, K., Qiu, M., Sun, X. et al. Smart data deduplication for telehealth systems in heterogeneous cloud

computing. J. Commun. Inf. Netw. **1**, 93–104 (2016). https://doi.org/10.1007/BF03391583.

[6] Kunjir, Ajinkya & Sawant, Harshal & Shaikh, Nuzhat. (2017). Data mining and visualization for prediction of multiple diseases in healthcare. 329-334. 10.1109/ICBDACI.2017.8070858.

[7] Sharmila, Leoni. (2022). A Comparative Study of Neural Network and Fuzzy Neural Network for Classification. Volume 10. 1371-78.

[8] Sarker, I.H. Machine Learning: Algorithms, Real-World Applications and Research Directions. SN COMPUT. SCI. **2**, 160 (2021). https://doi.org/10.1007/s42979-021-00592-x.

[9] S, Vinitha & S, Sweetlin & Vinusha, H.M. & S, Sajini. (2018). Disease Prediction Using Machine Learning Over Big Data. Computer Science & Engineering: An International Journal.8. 01-08. 10.5121/cseij.2018.8101.

[10] Mrs. K. Uma Maheswari, Ms. J. Jasmine, 2017, Neural Network based Heart Disease Prediction, International journal of engineering research & Technology (IJERT) RTICCT – 2017 (Volume 5 – Issue 17),

[11] Krishnaiah, V. & Narsimha, G. & Chandra, N.. (2015). Heart Disease Prediction System Using Data Mining Technique by Fuzzy K-NN Approach. Advances in Intelligent Systems and Computing. 337. 371-384. 10.1007/978-3-319-13728-5_42.

[12] Xiaoqing Li, Jiansheng Yang, Jinwen Ma,Recent developments of content-based image retrieval (CBIR),Neurocomputing,Volume 452,2021,Pages 675-689,ISSN,https://doi.org/10.1016/j.neucom.2020.07.139

[13] Gürbüz, Emre & Kilic, Erdal. (2013). A new adaptive support vector machine for diagnosis of diseases. Expert Systems. 31. 10.1111/exsy.12051.

[14] S. Narayanan, N. M. Balamurugan, M. K and P. B. Palas, "Leveraging Machine Learning Methods for Multiple Disease Prediction using Python ML Libraries and Flask API," 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 2022, pp. 694-701, doi: 10.1109/ICAAIC53929.2022.9792807.

[15] V. Sharma, S. Yadav and M. Gupta, "Heart Disease Prediction using Machine Learning Techniques," 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida, India, 2020, pp. 177-181, doi: 10.1109/ICACCCN51052.2020.9362842.