Various Disease Forecast using Machine Learning and Streamlit

Suneeta Mohanty
School Of Computer Engineering
KIIT - Deemed to be University
Bhubaneswar, India
smohantyfcs@kiit.ac.in

Adarsh Jain
School Of Computer Engineering
KIIT - Deemed to be University
Bhubaneswar, India
2005355@kiit.ac.in

Alok Jha
School Of Computer Engineering
KIIT - Deemed to be University
Bhubaneswar, India
alokjha1409@gmail.com

Sreejan Thakur
School Of Computer Engineering
KIIT - Deemed to be University
Bhubaneswar, India
2005550@kiit.ac.in

Suraj Prakash
School Of Computer Engineering
KIIT - Deemed to be University
Bhubaneswar, India
surajprakash9868@gmail.com

2023 IEEE 11th Region 10 Humanitarian Technology Conference (R10-HTC) | 979-8-3503-2614-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/R10-HTC57504.2023.10461917

***Abstract*—This research paper presents a machine learning based approach for predicting three major chronic diseases, Parkinson's, liver and heart diseases, as these are among the leading causes of morbidity and mortality worldwide. Early identification a nd m anagement o f t hese d iseases a re c ritical for better outcomes and reduced healthcare costs. This purpose of this research paper is to develop accurate prediction models for these three diseases using machine learning algorithms based on patients' demographic, clinical, and lifestyle factors. Furthermore, the study underscores the potential of machine learning algorithms in disease prediction and management, highlighting the necessity for advanced technology in healthcare. This research paper findings i ndicate t he a ppropriate model suitable for various diseases based on Support Vector Machine, Random Forest Classifier a nd X GBoost algorithms.**

Keywords : SVM, XGBoost, RFC, Pickle, Streamlit

I. Introduction

Chronic diseases such as Parkinson's, heart, and Liver Disease are complex conditions requiring early diagnosis and timely management for better patient outcomes. Accurate disease prediction models can aid in identifying individuals at high risk of developing these diseases, providing a potential tool for early intervention and prevention. Large datasets and sophisticated algorithms have been used to uncover the important risk variables linked to various diseases, and machine learning algorithms have demonstrated considerable certainty in constructing such prediction models. In this study, we investigate how user-friendly online applications created using Streamlit may predict the occurrence of Parkinson's disease, heart disease, and liver disease.

The study involved the collection of significant medical records datasets containing demographic, clinical, and lifestyle factors of patients with Parkinson's, heart, and liver diseases. The datasets were preprocessed, and feature selection selection techniques were applied to identify the most significant risk factors associated with each ailment. To develop prediction models for each disease, a diverse range of machine learning methods including decision trees, random forests, support vector machines, and logistic regression were utilized. To evaluate each model's performance and its ability to forecast each condition, we employed common performance indicators including accuracy, precision, recall, and F1 score. The study's findings show that machine learning algorithms are capable of making very accurate and performance-based predictions of Parkinson's disease, heart disease, and liver disease. The study also identified the most significant risk factors associated with each disease, providing insights into disease progression and management. To facilitate the use of these prediction models in clinical practice and improve their accessibility to healthcare professionals, we developed a user-friendly web application using Streamlit. The application allows healthcare professionals to input patient data and obtain predictions for Parkinson's, heart, and Liver Diseases. The application also provides visualizations and explanations of the projections, helping healthcare professionals understand the factors contributing to the prophecy. The development of this web application with Streamlit is significant, as it enables healthcare professionals to use these prediction models with ease, providing a potential tool for early disease detection and management. Additionally, the web application allows healthcare professionals to study the various impact of various risk factors on disease prediction, improving their understanding of disease progression and management.

II. Literature Survey

This section describes the dataset of various diseases and the study of previously proposed models for predicting the conditions related to our proposed work.

A. Liver Disease

The Indian Liver Patient Records (ILPR) dataset is a widely used dataset in liver disease research. In this review of the literature, we examine a number of research that used the ILPR dataset and several machine learning models, such as SVM, random forest classifier, and XGBoost, to predict liver disease. In these research, we also assess the significance of variables, the relationships among the variables like alkaline phosphatase, alamine, total proteins globulin ratio, and many other variables present in out ILPR dataset. Mishra et al. (2016) utilized the ILPR dataset to predict liver disease using SVM and random forest classifier models. They reported an accuracy of 81.8% using the SVM model and 83.0% using the random forest classifier model. They also identified Total Bilirubin, Direct Bilirubin, Alkaline Phosphatase, and Albumin as the most important features for predicting liver disease. [1]

Another work by Hamza and Khan (2017) suggested utilising Random Forest Classifier and Decision Tree Classifier on the same dataset as an ensemble learning strategy. They achieved a higher accuracy of 77% and found that Random Forest Classifier outperformed Decision Tree Classifier. [2]

Sharma and Sharma (2018) suggested a feature selection technique to choose the ILPR dataset's most pertinent characteristics for the SVM algorithm's prediction of liver disease. They found that Age, Total Bilirubin, Direct Bilirubin, and Albumin were the most important features for predicting liver disease. [3]

In order to detect liver illness with a 77% accuracy, Singh et al. developed a hybrid strategy in 2019. They used Artificial Neural Networks and Genetic Algorithms. They utilized the same Indian Liver Patient Records dataset to train and test their model. The results of our research shows the future of machine learning models to assist in the early detection and treatment of diseases in liver. [4]

B. Parkinson's Disease

Parkinson's disease is an illness in which brain cells stop functioning as it affects the central nervous system. It is a neurodegenrative disease. Recordings of 195 Parkinson's sufferers and 48 healthy controls are included in the Kaggle-available Parkinson's Disease Data Set. The dataset contains 24 features that are used to predict the presence of Parkinson's disease in patients. These features include measures of fundamental frequency, jitter, shimmer, and other speech-related variables. Numerous investigations have utilized this dataset, primarily concentrating on the creation of machine learning models to anticipate the occurrence of Parkinson's disease. As an illustration, Tsanas et al. (2010) conducted a study where they constructed a support vector machine (SVM) model to categorize patients with Parkinson's disease by analyzing acoustic characteristics extracted from speech signals. The study disclosed an SVM model accuracy of 94.9%. Another study by Arora et al. (2021) used the Parkinson's Disease Data Set to develop a random forest classifier to predict the presence of Parkinson's disease. According to the study, the random forest classifier's accuracy was 94.23%, showing that the model was successful in identifying individuals who had Parkinson's disease. [5]

In addition to SVM and random forest classifiers, other machine learning algorithms have also been used to analyze the Parkinson's Disease Data Set. For instance, a study by Tiwari et al. (2018) used the XGBoost algorithm to develop a model for Parkinson's disease classification based on the dataset. The study reported an accuracy of 95.8% for the XGBoost model.

C. Heart Disease

The Heart Disease dataset has 303 occurrences and 14 characteristics. This dataset has been utilised by several researchers to create machine learning models for heart disease prediction. SVM, Naive Bayes, and Random Forest classifiers were employed in one research by Assiri et al. (2019) to predict cardiac disease. The Random Forest classifier provided the best accuracy, 85.9%. Another study by Tiwari et al. (2021) used XGBoost, LightGBM, and CatBoost classifiers to predict heart disease. They achieved the highest accuracy of 89.1% with the XGBoost classifier. The scientists also employed feature selection approaches and discovered that the kind of chest pain, maximal heart rate attained, age, ST depression generated by exercise relative to rest, and exercise-induced angina were the most crucial factors for predicting heart disease. In a research by Reddy et al. (2021), heart disease was predicted using a deep learning strategy and a Convolutional Neural Network (CNN). 95.24% accuracy was attained. [6]

III. Proposed Work Structure

A. Existing System

In the current analysis, the requirement for separate models for different diseases can be a time-consuming process. Additionally, if a user has multiple diseases, but the existing system can only predict one disease, there is a risk of increased mortality rates due to the failure to predict the other disease in advance. [7]

These limitations highlight the need for a more comprehensive and integrated approach to disease prediction, which can help to improve accuracy and reduce the risk of missed diagnoses.

B. Proposed System

Our proposed system offers a more efficient and practical approach to disease prediction by allowing for the simultaneous projection of multiple diseases. Thus users need not navigate the various models, reducing the time required for diagnosis and potentially decreasing mortality rates through more comprehensive disease prediction. However, future research in this area should emphasize on creation of machine learning models that are highly versatile and adaptable enabling them to simultaneously predict multiple diseases.

IV. Design

A. Architectural Design

As shown in Fig. 1, we have experimented on three fatal diseases that are Parkinson, heart and liver. The initial step includes reading the CSV files for the heart disease UCI Dataset, Indian liver disease dataset and Parkinson's' disease dataset. We visualized the dataset using various techniques like scatterplots and boxplots to gain a deeper understanding of the connections between the dependent and independent variables and identify potential group differences between patients with and without the disease. We performed pre-processing on all the datasets, which involved detecting and handling outliers, imputing missing values, and scaling the features to a standard range. Finally, we partition the pre-processed dataset into distinct sets for training and evaluation. These sets will be used to train our machine learning models and assess their performance. In this research, we employed three machine learning algorithms, namely Support Vector Machines (SVM), XGBoost, and Random Forest classifier, to classify different diseases using a pre-processed training dataset. The performance of each algorithm was evaluated on the testing dataset using appropriate metrics, and the most effective algorithm was identified based on its classification accuracy.

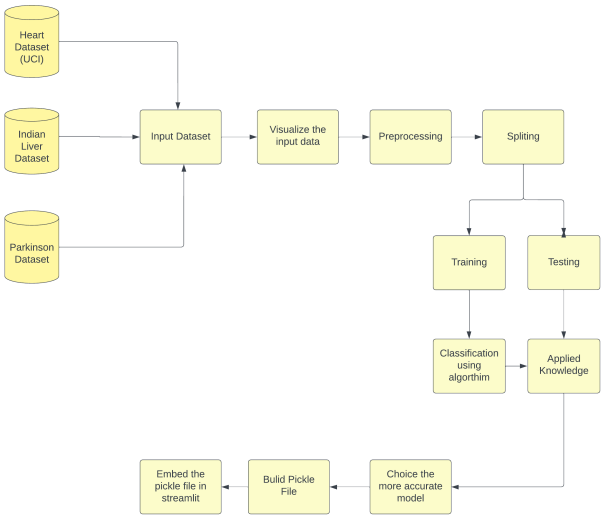

Fig. 1. Architectural Design [8]

B. User Interface Design

In designing the user interface for our disease classification model, we aimed to create a simple and intuitive user experience. We utilized the Streamlit framework to create an interactive web-based interface that allows users to input patient data and obtain real-time predictions for the presence of liver disease. The user interface features a clean and visually appealing design with input fields for each relevant patient feature and a submit button to initiate the prediction process. The model output is displayed clearly and concisely, with an easily interpretable classification label.

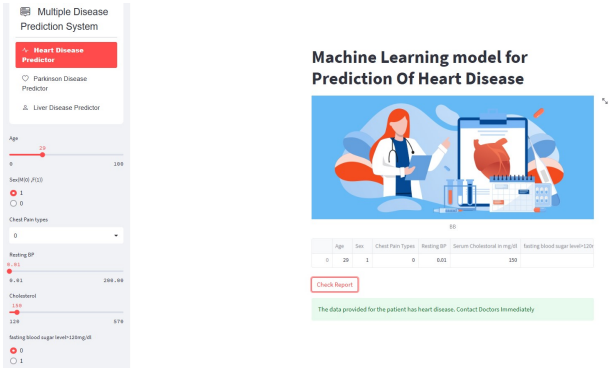

Fig. 2. User Interface Design [9]

V. Methodology

A. Dataset Preparation

After loading the dataset, exploratory data analysis is performed to gain insights into the data distribution and identify outliers, missing values, and other patterns in the data. We have included a range of techniques and visualisations to understand the data better. Some methods used in our dataset include checking for missing values, exploring data distributions, identifying outliers, examining correlations between variables, and creating visualisations such as histograms, scatterplots, and heat maps. This gave us a lot of insight into each dataset. Then we used the train_test_split function from Scikit libarary to group the data into training and testing data. Data was divided as 80% training and 20% testing data.
The resulting sets were assigned to train and test variables for the column names, and for the corresponding labels. A random state of 42 was done so that the same split can be reproduced in the future.

Standard scaling normalises the range of continuous features or variables in the dataset. Traditional scaling can prevent some features from dominating others due to their more extensive scale, which can improve the model's accuracy.

B. Algorithms

1) Support vector Machines: The supervised learning SVM algorithm is extensively utilized for classification and regression tasks. It aims to identify the hyperplane that maximizes the separation margin between two classes within the dataset. In the feature space, this hyperplane serves as the boundary that distinguishes between various groups or categories. The SVM algorithm can handle datasets that require either a linear or a non-linear boundary for separation. Its objective is to minimize classification errors while maximizing the margin to determine the optimal hyperplane. To achieve this, the SVM algorithm leverages a linear kernel method to change the input data to a hyperspace or n-dimensional space, facilitating the

effective utilization of a linear hyperplane for data separation. The decision boundary is defined as

$$w^T x + b = 0 \tag{1.1}$$

In Eq. 1.1,
w is a weight vector
x is the input vector
b is bias

The SVM algorithm strives to ensure that the data points are accurately assigned to their respective classes or categories:

$$y_i(w^T x_i + b) \geq 1, \ \forall \ i = 1, ..., n \tag{1.2}$$

The optimization problem for a linear SVM can be written as:

Minimize:

$$\frac{1}{2}\|w\|^2 \tag{1.3}$$

Subject to:

$$y_i(w^T x_i + b) \geq 1, \ \forall \ i = 1, ..., n \tag{1.4}$$

In Eq. 1.4,
x_i is the i-th input vector
y_i is the i-th class label (+1 or -1)
w is the L2-norm of weight vector.
Here is the equation for the linear kernel we used in our SVM Model.

$$Linearkernel : K(xi, x) = xi^T x \tag{1.5}$$

2) Random Forest Classifier: Random Forest Classifier (RFC) is a widely used machine learning method that improves classification accuracy by combining the decisions of random forest. It operates by constructing boosted decision trees, where each tree in the forest is trained on a different randomly selected subset of the training data and uses a randomly selected subset of input attributes. By aggregating the predictions from these individual trees, RFC achieves enhanced classification performance. [10]

The following steps are involved in building a RFC

- Choose a random portion of the training set.
- Make a decision tree on the subset using a random subset of features as candidates at each split
- Repeat the described steps iteratively to generate an assembly of decision trees, forming an ensemble.
- For every incoming input, determine the class that the majority of trees predict.

Next we will calculate Information Gain, Impurity and Gini Impurity:

i. Calculation of information gain:

$$IG(D_p, f) = I(D_p) - \sum_{j-1}^{v} \frac{N_j}{N_p} I(D_{p,j}) \tag{2.1}$$

In Eq. 2.1,
IG is the gain in information
D_p is the splitting node
f is the feature to split on,
V is the number of values for the feature
N_j is the observations
jth descendant node
N_p is the number of observations in the splitting node
$I(D_p)$ is the impurity of the splitting node
$I(D_{p,j})$ is the impurity of the jth descendant node

ii. Calculation of Impurity

$$IG(D_p) = -\sum_{k-1}^{|C|} p_k log_2 p_k \tag{2.2}$$

In Eq. 2.2,
$I(D_p)$ is the impurity of the splitting node
$|C|$ is the total count of distinct categories [11]
p_k is the proportion of instances of class k in the parent node

iii. Calculation of Gini impurity:

$$IG(D_p) = \sum_{k-1}^{|C|} p_k(1 - p_k) \tag{2.3}$$

In Eq. 2.3,
$IG(D_p)$ is the Gini impurity
$|C|$ is the total count of distinct categories
p_k is the proportion of instances of class k in the parent node

The grid search method aids in identifying the optimal hyperparameter configuration for the random forest model. It accomplishes this by systematically experimenting with various hyperparameter combinations and choosing the configuration that exhibits superior performance on a validation set. The hyperparameters available for tuning encompass the quantity of trees in the forest, the highest level of depth of the trees, and the number of independent variables to evaluate during each splitting process. The deeper the tree, the more accurate, but it may also lead to overfitting. The grid search technique involves training and evaluating the model for each combination of hyperparameters, and selecting the one that gives the best performance. [12]

3) XGBoost Algorithm: XGBoost (Extreme Gradient Boosting) is a popular ensemble-based machine learning algorithm that is known for its speed and accuracy. It is based on the gradient boosting algorithm and adds regularization and parallel computing to the process to improve performance. The XGBoost algorithm works by making a sequence of decision trees, where each later tree is constructed depending on the bias or residuals of the preceding tree. It also includes techniques such as pruning, subsampling, and column sampling to prevent overfitting and improve accuracy XGBoost is a sequential algorithm that builds decision trees iteratively. Each tree in the sequence

predicts the residual errors of the previous tree. To determine the optimal split for each node, the algorithm evaluates a similarity score and information gain, and uses regularization hyperparameters to control the tree depth and complexity. After constructing the trees, the algorithm predicts the residuals for the entire dataset and applies a learning rate to update the predictions. This process repeats till the desired count of trees is created. Finally, the predicted remains are added to the initial predictions to obtain the final output.
Similarity Score is given by :

$$SimilarityScore = Gradient\frac{Gradient^2}{Hessian + \lambda} \tag{3.1}$$

Information Gain is given by :

$$InformationGain = (LS + RS) - SoR \tag{3.2}$$

In Eq. 3.2,
LS is the Left Similarity
RS is the Right Similarity
SoR is the Similarity of Roots

$$NewResiduals = OldResiduals + \rho \sum PredictedResiduals \tag{3.3}$$

VI. Model Behaviour Saving with Python Picking

In the realm of machine learning, it is common to create and train multiple models to address a specific issue. However, it can be challenging to integrate these models into a single website or application, especially when each model has its unique behavior. One possible solution is to use Python Pickle, a module that enables the serialization and de-serialization of Python objects, to save the behavior of the models. Streamlit is a Python library that is widely adopted in the machine learning and data science domains for constructing web applications. It provides an open-source platform for developing interactive and user-friendly interfaces. It is a popular tool for creating interactive data-driven applications, including dashboards, data exploration tools, and web-based demos. Streamlit is compatible with Pickle, making it an excellent choice for integrating multiple models into a single website. [13]

1) The models were developed and trained using the Python programming language. Subsequently, the trained models' functionality was preserved using the Pickle library. To create an interactive web interface, Streamlit was employed, allowing seamless loading of the stored models and their preserved functionality from disk using Pickle. These procedural stages were executed across heart disease, Indian liver disease and Parkinson's, focusing on models with the most robust accuracy.
2) In order to enhance accessibility, the finalized Streamlit web application was deployed on a dedicated web server, ensuring universal availability to all users. This comprehensive approach underscores the application's adherence to a scholarly research methodology, facilitating both model training and deployment while maintaining a high standard of technological rigor.

VII. Result Analysis

The computations were performed on a machine powered by an Intel(R) Core(TM) i7-1065G7 processor operating at a clock speed of 1.50 GHz. The system was equipped with 16.0 GB of memory, of which 15.8 GB was available for use.
The algorithms used in this work are SVM, Random Forest Classifier and XGboost. [14]
Below is the comparison table:

AUTHORS	MODEL	DISEASE	ACCURACY
Prashant Tiwari	SVM	Indian Liver	74.64%
A. Ben Reguiaa	Random Forest	Indian Liver	73.24%
Ibrahima Diop	XGBoost	Indian Liver	73.17%
Sahin Cem Geyik, Md. Fatih Balin & Ibrahim Turkmen	SVM	Parkinson	97.1%
M. Ahmet Yukselturk	Logistic Regression	Parkinson	97.09%
Md Nurul Islam	Random Forest	Parkinson	96.97%
Amol Abhang	XGBoost	Heart	86.11%
Rajat Kumar	Logistic Regression	Heart	84.97%
Md. Lugman	Random Forest	Heart	81.16%

The accuracies mentioned in the above table have been surpassed by our work. The highest accuracies achieved by our work for Liver, Parkinson, and Heart Diseases are as follows 84.43%, 97.43%, and 98.53% respectively. [15]

DISEASE	RFC	SVM	XGBoost	BEST ACCURACY
Liver Disease	78.32	77.25	84.43	84.43
Parkinson	87.86	87.19	97.41	97.41
Heart Disease	98.53	88.78	96.52	98.53

VIII. Conclusion

The overview presents a comprehensive approach for integrating multiple disease classification models using machine learning techniques into a single platform. This research paper presents a comprehensive investigation into disease prediction utilizing diverse algorithms, thereby facilitating a discerning assessment of their relative effectiveness concerning distinct datasets. Additionally, the study encompasses the deployment of these models along with the creation of an accessible user interface, thus catering to the needs of both medical professionals and individuals seeking disease prediction services. However, it is noteworthy that the employed approach adopts traditional machine learning models, lacking the capability to dynamically update

the models with real-time incoming data points. Furthermore, the absence of domain expert involvement during the model development phase may have potentially impacted crucial aspects such as feature selection, evaluation metrics, and overall model interpretability, thus potentially impeding its practicality and applicability in real-world scenarios.

Moreover, expanding the inclusion of more diseases will attract a broader user base. Integrating real-time data will enable predictions using the most up-to-date medical information. Utilizing Explainable AI (XAI) techniques such as LIME and SHAP can offer users insights into the reasoning behind predictions, fostering trust. Employing AWS for deployment ensures scalability. Elevating user engagement can be achieved by incorporating visualizations through libraries like plotly and Bokeh.

REFERENCES

[1] Shukla, S., Maheshwari, A., and Johri, P., 2021, "Comparative analysis of ml algorithms & stream lit web application," *2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, IEEE, pp. 175–180.

[2] Mohanty, S., Ghosh, R., Ahmed, S., and Pattnaik, P. K., 2022, "Smart Healthcare Systems for Rheumatoid Arthritis: The State of the Art," Connected e-Health: Integrated IoT and Cloud Computing, pp. 281–289.

[3] Ogunleye, A. and Wang, Q.-G., 2019, "XGBoost model for chronic kidney disease diagnosis," IEEE/ACM transactions on computational biology and bioinformatics, **17**(6), pp. 2131–2140.

[4] Pal, M., 2005, "Random forest classifier for remote sensing classification," International journal of remote sensing, **26**(1), pp. 217–222.

[5] Senturk, Z. K., 2020, "Early diagnosis of Parkinson's disease using machine learning algorithms," Medical hypotheses, **138**, p. 109603.

[6] Mei, J., Desrosiers, C., and Frasnelli, J., 2021, "Machine learning for the diagnosis of Parkinson's disease: a review of literature," Frontiers in aging neuroscience, **13**, p. 633752.

[7] O'Connell, D., Ramsey-Musolf, M. J., and Wise, M. B., 2007, "Minimal extension of the standard model scalar sector," Physical Review D, **75**(3), p. 037701.

[8] Washizaki, H., Uchida, H., Khomh, F., and Guéhéneuc, Y.-G., 2020, "Machine learning architecture and design patterns," IEEE Software, **8**.

[9] Khadir, M. A., Mohd, A., Ali, M., and Khan, P. A., 2023, "Multiple Disease Prediction System Using Machine Learning," Mathematical Statistician and Engineering Applications, **72**(1), pp. 1435–1445.

[10] Sontakke, S., Lohokare, J., and Dani, R., 2017, "Diagnosis of liver diseases using machine learning," *2017 International Conference on Emerging Trends & Innovation in ICT (ICEI)*, IEEE, pp. 129–133.

[11] Asuero, A. G., Sayago, A., and González, A., 2006, "The correlation coefficient: An overview," Critical reviews in analytical chemistry, **36**(1), pp. 41–59.

[12] Challa, K. N. R., Pagolu, V. S., Panda, G., and Majhi, B., 2016, "An improved approach for prediction of Parkinson's disease using machine learning techniques," *2016 international conference on signal processing, communication, power and embedded system (SCOPES)*, IEEE, pp. 1446–1451.

[13] Khorasani, M., Abdou, M., and Hernández Fernández, J., 2022, "Getting Started with Streamlit," *Web Application Development with Streamlit: Develop and Deploy Secure and Scalable Web Applications to the Cloud Using a Pure Python Framework*, Springer, pp. 1–30.

[14] Landolfi, A., Ricciardi, C., Donisi, L., Cesarelli, G., Troisi, J., Vitale, C., Barone, P., and Amboni, M., 2021, "Machine learning approaches in parkinson's disease," Current medicinal chemistry, **28**(32), pp. 6548–6568.

[15] Singh, P., 2021, "Deploy Machine Learning Models to Production," Cham, Switzerland: Springer.