JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

MediXpert : Multiple Disease Diagnosis System

Akshay Oza
Computer Engineering SAKEC
Mumbai, India akshay.oza16479@sakec.ac.in

Shreyash Chaukhande
Computer Engineering SAKEC
Mumbai, India shreyash.chaukhande16150@sakec.ac.in
Aniket Jadhao *Computer Engineering SAKEC*
Mumbai, India aniket.jadhao16149@sakec.ac.in

Karuna Borhade
Assistant Professor, Computer Engineering SAKEC
Mumbai, India karuna.borhade@sakec.ac.in
Shivraj Ugale *Computer Engineering SAKEC*
Mumbai, India shivraj.ugale16455@sakec.ac.in

***Abstract*—Increasing incidence of different diseases emphasizes the need to develop accurate predictions. Such models assist in detecting issues earlier and initiating treatment sooner. In this post, the performance of six machine learning models: Logistic Regression, Random Forest, Support Vector Machine (SVM), Decision Tree, & Gradient Boosting in predicting various diseases is analyzed. Clinical & demographic info from patients suffering from an in depth dataset, & many other diseases. The models used were trained and tested using a method known as 10-fold cross validation. We evaluated the performance of these algorithms using various performance indicators such as accuracy, precision, recall, & F1-score. Processing this data revealed the Gradient Boosting algorithm to yield the highest accuracy, which was 93.2%. Additioanlly, Random Forest & SVM exhibited close to similar good performances This almost shows that we train machine learning algorithms to predict many diseases or conditions. These outcomes imply Gradient Boosting, Random forest & SVM is an appropriate disease prediction technique and can be helpful for developing reliable clinical based predictive models.**

***Index Terms*—Multiple disease prediction, machine learning, logistic regression, random forest, SVM, decision tree, gradient boosting.**

I. INTRODUCTION

The swift evolution of medical technologies and growing accessibility of health-related information has unlocked fresh possibilities for enhancing illness identification and therapeutic approaches. Yet healthcare practitioners face substantial hurdles due to the intricate nature of human ailments and the diverse ways individuals respond to interventions. Recently, computational learning systems have emerged as a powerful tool for strengthening diagnostic accuracy and treatment strategies. These sophisticated systems can analyze enormous collections of information, recognize subtle relationships, and generate accurate forecasts. The growing prevalence of various illnesses has become a pressing global challenge, touching the lives of millions battling diverse health complications. Conditions such as blood sugar imbalances, heart-relatedailments, progressive neurological tremor disorders, ongoing renal dysfunction, compromised liver function, and viral liver inflammation stand out as particularly widespread and de- bilitating health threats worldwide. Beyond diminishing life quality, these medical conditions impose substantial financial hardships on patients, their family circles, and healthcare infrastructure systems. The complexity of multiple disease prediction lies in the presence of multiple factors, including genetic, environmental, and lifestyle factors. These factors are interconnected in intricate ways, making it difficult to determine the root causes of diseases. Moreover, the symptoms of multiple diseases can be similar, making it difficult to diagnose and treat them accurately. Traditional diagnostic methods, such as laboratory tests and physical examinations, have limitations in detecting multiple diseases simultaneously. These methods are often time-consuming, expensive, and may not provide accurate results. Additionally, they may struggle to detect intricate patterns and correlations among different health indicators. In recent years, computerized prediction sys- tems have revealed tremendous promise in forecasting various health conditions, including diabetes, heart disease, and cancer. These digital frameworks can process enormous datasets, uncover subtle patterns, and generate accurate predictions. By harnessing these computational intelligence tools, healthcare providers can develop predictive systems that identify at-risk patients, enable early condition detection, and suggest cus- tomized

treatment paths. Building highly accurate forecasting models for multiple condition identification requires analyzing extensive information from diverse sources. Learning algorithms such as Logistic Regression, Random Forest, Support Vector Machine (SVM), Decision Tree, and Gradient Boosting have been widely applied for illness prediction. However, these models' effectiveness depends on factors like data quality, condition complexity, and algorithm selection.

II. Background

Many existing healthcare analysis systems are limited to predicting a single disease. For instance, one model may analyze diabetes, another diabetic retinopathy, and another heart disease. Most current systems are specific to individual conditions, so when organizations need to review health reports for their patients, they often need multiple models. This singledisease approach is restrictive and lacks efficiency for broader health analysis. A multidisease prediction system, by contrast, allows users to check for various conditions on a single platform, removing the need to visit multiple sites.

The main aim of developing such a system is to support doctors in verifying their diagnoses, ultimately contributing to improved healthcare outcomes. Our work aims to create a unique, versatile platform for early and accurate multi-disease diagnosis. Existing systems often suffer from reduced accuracy when medical data is incomplete. Additionally, regional variations in disease characteristics can sometimes lead to incorrect predictions. To address these challenges, our proposed approach leverages machine learning and Convolutional Neural Networks (CNNs) to provide precise disease predictions, resulting in a more reliable and effective diagnostic solution.

III. Problem Statement and Objectives

Problem Statement

Many current machine learning models in healthcare focus on predicting a single disease at a time. For instance, some models are designed exclusively for liver analysis, while others target cancer or lung diseases. As a result, users seeking to predict multiple diseases must navigate various platforms, as there is no integrated system capable of performing multi-disease predictions in a single analysis. Additionally, some of these models demonstrate low accuracy, which can adversely impact patient health outcomes. When organizations analyze patient health reports, they often need to implement multiple models, leading to increased costs and longer processing times. Furthermore, several existing systems rely on a limited number of parameters, which may result in inaccurate predictions.

A. Objectives

The objective of *Medi-Xpert* is to overcome the challenges of web-based information retrieval by achieving the following:

- In multiple disease prediction system, users can forecast more than one disease simultaneously, eliminating the need to visit different platforms for each prediction. Our focus is on three interrelated conditions: liver disease, diabetes, and heart disease. This approach enables a comprehensive analysis of these interconnected diseases.
- Develop a multi-scale hybrid architecture for multiple disease prediction using Na¨ıve Bayes, K-Nearest Neigh- bors (KNN), and Support Vector Machine (SVM), this approach replaces CNNs and Vision Transformers with traditional machine learning algorithms adapted for the detection of different disease patterns.

IV. This Architecture will incorporate multi-scale feature extraction and dynamic classifier selection to enhance interpretability and achieve high accuracy.

Literature Survey

The increasing adoption of machine learning (ML) techniques in the healthcare domain has opened promising avenues for early disease detection and prediction. Researchers have developed numerous approaches leveraging both traditional algorithms and deep learning models across diverse disease categories. This literature survey presents a unified flow of research work focused on multiple disease prediction using ML, structured to reflect continuity and thematic progression. Anurag Dharwal et al. [1] proposed a robust ML-based framework for the early prediction of Chronic Kidney Disease (CKD), utilizing demographic, clinical, and laboratory data. The study emphasizes the importance of early diagnosis to delay disease progression and incorporates a combination of feature selection and classification algorithms. Their approach

achieved an accuracy of 89

Following the study on chronic diseases, Tahira Islam Trishna et al. [2] focused on the detection of hepatitis types A, B, C, and E using Random Forest, K-nearest Neighbor (KNN), and Na¨ıve Bayes classifiers. Utilizing the WEKA software, the classification algorithms were evaluated, with Random Forest achieving the highest accuracy of 98.6

In a similar vein, Selamawit Sileshi Nigatu et al. [3] presented a comparative study on liver disease prediction using supervised learning algorithms, including Random Forest, De- cision Tree, Support Vector Classifier, K-Nearest Neighbour, AdaBoost, Stochastic Gradient Descent, and Artificial Neural Network (ANN). Utilizing the Indian Liver Patient Dataset, the study reported that ANN was the most effective, achieving an accuracy of 87

Complementing the research on chronic diseases, another study titled "Research of Heart Disease Prediction Based on Machine Learning" [4] investigates the use of various algo- rithms such as Decision Trees, Random Forests, and Support Vector Machines (SVM) for predicting heart disease. The paper emphasizes the potential of these methods in classifying risk and enabling early clinical intervention. This contributes to the broader goal of integrating heart disease prediction into multi-disease models.

Transitioning to acute disease prediction, Massimo Giotta et al. [5] employed a Decision Tree model to predict outcomes for non-intensive COVID-19 inpatients. Their prospective study revealed that the international normalized ratio (INR) and immunoglobulin M (IgM) were critical predictive features. The model achieved 75.93

For lightweight and user-friendly applications, Nitesh Kumar Verma et al. [6] developed a disease prediction system using the Na¨ıve Bayes classifier. Based on symptom inputs, the model calculates disease probabilities and predicts the most likely outcome. While the model

achieved an average prediction accuracy of 60

Alvin Rajkomar et al. [7] explored scalable and accurate deep learning applications on electronic health records (EHRs), transforming raw clinical data into structured inputs using the Fast Healthcare Interoperability Resources (FHIR)

format. Their deep learning models demonstrated superior performance across multiple tasks, including in-hospital mortality (AUROC 0.93–0.94), 30-day readmission (0.75–0.76), and length of stay (0.85–0.86). These models significantly outperformed traditional methods, emphasizing the value of unstructured EHRs in training high-performance multi-disease models.

Focusing on oncology, Konstantina Kourou et al. [8] reviewed machine learning techniques applied to cancer prognosis and prediction. The study discusses the use of Support Vector Machines (SVM), Artificial Neural Networks (ANN), Decision Trees (DT), and Bayesian Networks (BN) in predicting cancer susceptibility, recurrence, and survival. It highlights the trend of integrating genomic and clinical data, stressing the need for model validation before clinical implementation. The review confirms that ML applications can significantly enhance cancer diagnostics by improving accuracy in outcome prediction.

Providing essential data infrastructure for large-scale ML research, Cathie Sudlow et al. [9] introduced the UK Biobank—a population-based cohort comprising over 500,000 participants aged 40–69. The dataset includes genetic, environmental, and clinical data, enabling wide-ranging studies across multiple disease categories. The open-access design supports the development of diverse predictive models and facilitates disease correlation analysis over long follow-up periods.

Similarly, Alistair E. W. Johnson et al. [10] presented the MIMIC-III database, a freely accessible critical care dataset containing de-identified health data from over 53,000 ICU admissions. It includes demographics, vital signs, medications, and laboratory measurements, making it a crucial resource for training ML models on multi-system disease prediction, especially in critical care contexts.

V. Limitations of Existing Systems

The reviewed research demonstrates significant advancements in the application of machine learning for disease prediction, several common limitations have been observed across the studies. These limitations, ranging from dataset constraints and algorithmic challenges to issues with model evaluation and clinical applicability, highlight areas that require further attention to enhance the reliability, scalability, and generalizability of predictive systems in real-world healthcare environments.

A. Dataset Limitations

Limited Dataset Size and Scope The CKD study used a relatively small dataset with limited diversity in demographic and clinical features, which affects generalizability to different healthcare settings [1].

Lack of Dataset Diversity The liver disease prediction study was restricted to the Indian Liver Patient Dataset (ILPD), which does not represent global populations and might not generalize to different regions [3].

Single-Hospital Source The COVID-19 prediction model used data from a single Italian hospital, introducing potential regional or institutional bias [5]. Similarly, the MIMIC-III dataset is based solely on ICU admissions from one hospital, which limits its application to broader or non-critical care scenarios [10].

Ethnic Homogeneity The UK Biobank consists mostly of dividuals of European descent from the UK, reducing appli- cability to ethnically diverse populations [9].

B. Algorithmic and Model Limitations

Lack of Model Complexity The Na¨ıve Bayes model for general disease prediction employed a very basic algorithm with limited prediction accuracy (60

Omission of Ensemble/Deep Learning Approaches The liver disease study, while comparative, did not test ensemble or deep learning models, which could have improved performance [3]. Black Box Nature of Deep Learning The deep learning models developed using EHRs showed excellent performance but lacked interpretability, making it difficult to trust or explain predictions in clinical environments [7].

C. Evaluation and Validation Limitations

Limited Performance Metrics The hepatitis detection paper used only accuracy as the performance metric, omitting essen- tial clinical metrics like precision, recall, specificity, and AUC [2].

Lack of External Validation Multiple studies, particularly the heart disease prediction paper and the cancer prognosis re- view, lacked external dataset validation or deployment in real- world clinical settings, affecting credibility and reproducibility [4][8].

D. Feature and Preprocessing Limitations

Incomplete Feature Engineering The hepatitis virus detection study did not detail preprocessing steps or feature engineering, which are crucial in medical ML models [2].

Omitted Feature Selection Details The heart disease study did not report on feature selection, data balancing, or prepro- cessing, making replication or improvement difficult [4].

E. Clinical Applicability and Scalability

Clinical Applicability and Scalability Low Specificity in Clinical Deployment Although the COVID-19 prediction model had high sensitivity, its low specificity (23.43

High Resource Requirements The deep learning models trained on full EHRs require significant computational power and data infrastructure, limiting deployment in smaller health- care settings [7].

VI. Proposed System

A. Feasibility Study

To overcome the limitations of isolated diagnostic tools and deliver a unified intelligent healthcare platform, a comprehen- sive feasibility study was conducted. This study evaluated the technical, operational, and economic viability of implementing a scalable multi-disease prediction system using pre-trained machine learning models specifically for five critical diseases: diabetes, liver disease, kidney disease, hepatitis C, and heart disease.

1) Integration of Specialized ML Models::

- Model Availability: The system utilizes pre-trained

.sav machine learning models for each of the five target diseases. These models are loaded dynamically using Python's pickle library, enabling a modular and main- tainable architecture.

- Disease-Specific Training: Each model is trained on clin- ically relevant datasets containing disease-specific fea- tures, enhancing diagnostic precision and reducing false positives.

2) Data Ingestion and Prediction Handling::

- Dynamic Input Routing: The system accepts structured patient inputs through a unified interface and preprocesses them to match the format required by the selected disease model.
- Backend Prediction Mechanism: A centralized prediction handler dynamically loads the correct .sav model based on disease selection and generates real-time predictions using NumPy-based input reshaping and predict() execu- tion.

3) Scalability and Performance:

- Independent Execution Pipelines: Each disease model runs within its own processing module, allowing for concurrent predictions and independent scalability.
- Low Latency and High Throughput: Preloading of mod- els and efficient I/O operations ensure quick diagnostic feedback suitable for clinical use cases.

4) Open-Source and Customizable::

- Interoperable Model Files: The .sav format provides flex- ibility to retrain, replace, or update models using open- source libraries like Scikit-learn.
- Adaptable Interface: The user interface can be easily ex- tended to include new disease models or input parameters without structural changes to the backend.

5) Autonomous Risk Evaluation::

- Interactive User Dashboard: A simple and intuitive GUI allows users to select the target disease and enter their clinical parameters. The relevant model is invoked based on the selection.
- Readable Results: Binary or multi-class model outputs are converted into meaningful interpretations (e.g., "Risk of Heart Disease Detected" or "No Hepatitis Indicated").

6) System Maintenance and Updates:

- Model Hot-Swapping: Developers or medical researchers can independently retrain and update individual models, enhancing maintainability.
- Version Tracking: System logs maintain version histories of each model to ensure reproducibility and traceability of diagnostics..

7) Cost-Effective Solutions:

- Offline Execution: The system performs all predictions locally using lightweight .sav models, avoiding depen- dency on cloud services and reducing operational costs.
- Minimal Third-Party Requirements: The solution is im- plemented primarily in Python using Scikit-learn and NumPy, minimizing licensing and infrastructure costs.

8) Efficient Resource Utilization::

- As-Needed Model Execution: Only the selected model is loaded during a diagnostic session, reducing computa- tional overhead.
- Lightweight Architecture: Designed for low- resource en- vironments, the system can run effectively even on stan- dard hardware.

B. System Architecture

The proposed system adopts a modular and extensible architecture tailored for multi-disease prediction. It focuses on five major diseases—Diabetes, Liver Disease, Kidney Disease, Hepatitis C, and Heart Disease—utilizing disease-specific machine learning models, each encapsulated in a sep- arate processing module. The architecture ensures independent execution, seamless integration, efficient data handling, and real-time diagnostic capability.

The architecture comprises several interconnected compo- nents working in harmony to deliver accurate predictions.

At the forefront is the User Interface Layer, which serves as the interaction point between the user and the prediction system. This layer allows the user to select the disease category from a dropdown list and input the required clinical attributes. A streamlined graphical interface ensures usability for both medical professionals and patients, incorporating input valida- tion mechanisms to prevent incorrect submissions. The output is presented in natural language, such as "No risk of Diabetes" or "Potential signs of Liver Disease detected."

Next is the Input Preprocessing Module, which validates, cleans, and transforms the user input into a machine-readable format. The entered values are converted into a NumPy array and reshaped to a 2D structure using reshape(1, -1) to meet the model's expectations. Any missing or malformed inputs are handled using default values or flagged for user correction. Additionally, this module ensures that the order and number of input features align precisely with the model's training schema.

The Model Selection and Loading Engine follows, acting as a dynamic router that selects and loads the appropriate model based on the user's disease selection. Once the user selects a disease, the engine identifies the corresponding .sav file from the model repository. Using Python's pickle library, the model file is deserialized to create an executable ML object. For efficiency, frequently used models can optionally be preloaded into memory, reducing I/O overhead and enhancing response times.

At the core of the architecture lies the Prediction Engine, which is responsible for producing the final disease prediction. It receives the processed feature vector along with the loaded model instance and executes the model's .predict() function on the reshaped input. The engine outputs a binary or multi-class label indicating the presence or absence of disease.

Complementing the engine is the Output Interpretation and Feedback Module, which translates the model's raw output

into meaningful clinical interpretations. Numerical results such as 0 or 1 are mapped to understandable labels like "No Disease" or "At Risk." If a risk is detected, the system can optionally generate alert messages prompting users to seek medical advice. Outputs are formatted concisely and in non-technical language to enhance user comprehension.

Supporting these components is the Logging and Versioning System, which, while optional, is critical for model version

control and diagnostic reproducibility. It tracks which version of each model was used for a given prediction and stores timestamped user inputs and results for system audits or potential clinical studies.

Finally, the Backend Integration Layer acts as middleware, connecting the frontend UI with the prediction logic and the file system. Built in Python, it integrates libraries such as NumPy, Scikit-learn, and Pickle to ensure seamless function-ality. This layer handles secure and efficient access to model files stored in local directories or cloud buckets and manages the sequential workflow from input to prediction output

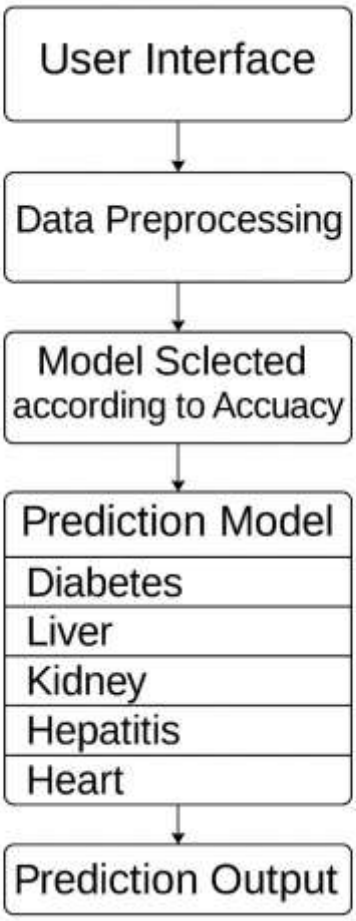

Fig. 1. Flowchart of System Architecture

This flowchart represents the overall workflow of a multi-disease prediction system. It begins with the User Interface, where users input their clinical data. The Data Preprocessing step then validates and formats the input to ensure compatibil- ity with the machine learning models. Next, the system dynam- ically selects the best-performing model based on accuracy in the step labeled Model Selected according to Accuracy (note: "Slected" should be corrected to "Selected" and "Accucay" to "Accuracy").

After selection, the Prediction Model specific to the dis-ease (Diabetes, Liver, Kidney, Hepatitis, or Heart) is applied. Finally, the Prediction Output displays the diagnostic result, helping users or medical professionals interpret the health status based on model predictions.

VII. Implementation and Results

A. Implementation Details

Medixpert is a comprehensive health prediction platform that analyzes both structured and unstructured medical data to predict multiple diseases. It targets the prediction of Diabetes, Heart Disease, Parkinson's Disease, Chronic Kidney Disease, Liver Disease, and Hepatitis using six advanced machine learning algorithms:

• Logistic regression • Random forest • Support vector machine (SVM) • Decision tree • Gradient boost.

By integrating structured data from patient records and unstructured data such as medical reports , Medixpert aims to provide early, accaccurate reliable disease risk assessments.

4.1 Data Collection:

Structured Data: Structured medical datasets were collected from verified online repositories (such as Kaggle, UCI Ma-chine Learning Repository). Key features (e.g., blood pressure, glucose levels, cholesterol) were selected based on medical relevance.

Unstructured Data: Unstructured data such as medical notes, discharge summaries, and health articles were also sourced from trusted health websites and clinical reports.

Authenticity: All symptoms and disease-related information were sourced carefully without the inclusion of dummy values. Only real, medically validated symptoms and markers were considered, ensuring the dataset's reliability and authenticity.

4.2 Data Preprocessing :

Prior to modeling, a comprehensive data cleaning and preprocessing phase was performed:

Handling Missing Values: Missing data in structured datasets were addressed using the Latent Factor Model and forward fill methods, ensuring that imputation preserved data integrity.

Data Standardization: All numerical features were stan-dardized (mean = 0, standard deviation = 1) to normalize feature scales, benefiting algorithms like Logistic Regression and SVM.

Data Splitting: The dataset was divided into training and testing sets (typically 80-20 ratio split to allow model valida- tion on unseen data.)

Feature Selection: For structured data, important features were manually selected. For unstructured text files, the Ran- dom Forest Algorithm was used to automatically select the most relevant features by analyzing feature importance.

• 4.3 Model Building:

Machine learning models were trained using various classi-fication techniques to handle multi-disease prediction.

Algorithms Implemented:

- Random Forest
- Logistic Regression
- Support Vector Machine (SVM)
- Decision Tree
- Gradient Boosting

Techniques Used: Grouping and clustering for initial data exploration. Summarization for understanding key patterns in data. Hyperparameter tuning (Grid Search and Random Search) to optimize each model. 5-Fold Cross-Validation to avoid overfitting.

Classification Focus: Classification tasks were emphasized

since disease prediction involves assigning categorical outcomes (e.g., disease present = 1, disease absent = 0).

4.4 Prediction:

Each disease was mapped to the best-performing machine learning models based on evaluation results:

Diabetes: Random Forest

Heart Disease: Random Forest, Decision Tree Chronic

Kidney Disease: Random Forest

Liver Disease: Logistic Regression, Random Forest. Hepatitis: Support Vector Machine (SVM), Gradient Boosting, Support Vector Machine (SVM)

Each model was tested for metrics like accuracy, precision, recall, F1-score to ensure robust performance.

4.5 Algorithm Descriptions

4.5.1 Random Forest

Algorithm Working Principle:

Step 1: Random subsets (samples) are selected from the training dataset.

Step 2: Independent decision trees are built from each subset.

Step 3: A pre-determined number of decision trees (N) are generated.

Step 4: For new data, each tree votes on the prediction.

Step 5: The final prediction is made based on the majority vote (classification) or averaged results (regression).

Strengths: Reduces overfitting.

Works well with both categorical and numerical data.

4.5.2 Support Vector Machine

(SVM) Working Principle:

Step 1: SVM finds the optimal hyperplane that separates data points of different classes.

Step 2: Converts the problem into an optimization problem using the hinge loss function.

Step 3: Balances margin maximization with minimizing classification error using a regularization parameter.

Step 4: Solves using gradient-based optimization techniques.

Step 5: Final classification is made based on which side of the hyperplane a new data point falls.

Strengths: Effective in high-dimensional spaces.

Works well with clear margin separation between classes.

4.5.3 Logistic

Regression Working

Principle:

Step 1: Initialize model parameters (weights and bias).

Step 2: Apply the logistic (sigmoid) function to predict robabilities for the output class.

Step 3: Define a cost function (typically cross-entropy loss) to measure prediction error.

Step 4: Optimize the cost function using techniques like gradient descent to update model parameters.

Step 5: For new data, predict class based on whether the output probability is above or below a set threshold (e.g., 0.5).

Strengths: Simple and easy to implement.

Provides probabilistic interpretation of predictions.

Works well when the relationship between input variables and the output is linear.

4.5.4 Decisio

n Tree Working

Principle:

Step 1: Select the best feature to split the dataset based on criteria like Gini impurity, entropy (information gain), or variance reduction.

Step 2: Split the dataset into subsets based on the selected feature's values.

Step 3: Repeat Step 1 and Step 2 recursively for each subset until a stopping condition is met (e.g., maximum depth, minimum samples).

Step 4: Create leaf nodes that predict the final output (class or value).

Step 5: For new data, traverse the tree from the root to a leaf based on feature values to make a prediction.

Strengths: Easy to interpret and visualize.

Handles both numerical and categorical data.

Requires little data preprocessing (no need for feature scaling).

4.5.5 Gradient

Boosting Working

Principle:

Step 1: Start with an initial simple model (often predicting the mean value for regression or a constant probability for classification).

Step 2: Compute the error (residuals) of the current model's predictions.

Step 3: Train a new weak learner (usually a small decision tree) to predict the residuals.

Step 4: Update the model by adding the new weak learner's predictions, scaled by a learning rate.

Step 5: Repeat Steps 2–4 for a specified number of iterations or until the error is minimized.

Step 6: For new data, make predictions by summing the outputs of all weak learners.

Strengths: Produces highly accurate models.

Can handle mixed types of data and missing values.

Effective at capturing complex non-linear rela

4.6 Technologies and Tools

Used Programming Language:

Python

Libraries and Frameworks: Scikit-learn (machine learning models), Pandas and NumPy (data manipulation), Matplotlib and Seaborn (data visualization), Streamlit (for building web application).

Development Environment: Jupyter Notebook, VS Code, Google Colab

B. Results

In the disease prediction system, distinct machine learning algorithms are tailored to specific diseases based on their performance and accuracy. For instance, the Random Forest algorithm is utilized for Diabetes and Chronic Kidney Disease prediction, as it demonstrated high accuracy in these domains. For Heart Disease, both the Random Forest and Decision Tree algorithms are applied, capitalizing on their strong predictive performance. In the case of Liver Disease, a combination of Logistic Regression and Random Forest is employed to achieve more robust and reliable results. For Hepatitis, Support Vector Machine (SVM) and Gradient Boosting algorithms are used, with SVM particularly favored due to its consistent accuracy, while Gradient Boosting enhances overall model performance.

The initial performance results are summarized below:

Disease	Algorithm	Accuracy
Heart Disease	Decision Tree	80%
	Random Forest	87%
Diabetes	Random Forest	99%
Hepatitis	Logistic Regression	90.24%
	Random Forest	92.68%
	Gradient Boosting	93.50%
	Support Vector Machine (SVM)	90.24%
Liver Disease	Random Forest	67.48%
	Logistic Regression	66.67%
Chronic Kidney Disease	Random Forest	100%

Fig. 2. Inital Accuracy

Model	Algorithm	Accuracy
Diabetes Diseases	Random Forest Algorithm	94.90%
Heart Diseases	Random Forest Algorithm	87%
Lever Diseases	Random Forest Algorithm	67.48%
Hepatitis Diseases	Gradient Boosting	93.50%
Chronic Kidney Diseases	Random Forest Algorithm	100%

Fig. 3. Accuracy

When a patient inputs parameters corresponding to a particular disease, the system evaluates the likelihood of disease occurrence based on the provided information. Each parameter is associated with a predefined valid value range to ensure data relevance and accuracy.

If any entered values are outside the specified range, invalid, or left empty, the system alerts the user with a warning prompt to correct the input.

This input validation mechanism ensures high-quality data, resulting in more accurate and reliable predictions tailored to individual patient profiles and specific disease categories.

Furthermore, if the system predicts that the patient is likely affected by a disease, it offers an additional feature to set eminders for medication timings (e.g., pill intake schedules). This feature is especially provided for diseases such as:

Diabetes
Heart
Disease
Chronic Kidney Disease
Liver Disease
Hepatitis

In total, five machine learning algorithms — Logistic Regression, Random Forest, Support Vector Machine (SVM), Decision Tree, and Gradient Boosting — are employed for predicting multiple diseases, with the best-performing model chosen for each disease based on accuracy and clinical rele- vance.

User-Centric Interface :

Fig. 4. Diabetes Interface

Fig. 5. Heart Interface

Fig. 6. Chronic Interface

VIII. CONCLUSION

In conclusion, The MediXpert system successfully demonstrates the power of machine learning in healthcare by accurately predicting multiple diseases, including Diabetes, Heart Disease, Chronic Kidney Disease, Liver Disease, and Hepatitis. By leveraging advanced algorithms such as Random Forest, Support Vector Machine (SVM), Logistic Regression, Decision Tree, and Gradient Boosting, the system achieved high accuracy levels, ensuring reliable and timely disease detection.

MediXpert not only simplifies the diagnostic process by providing users with a single, integrated platform but also reduces the time, effort, and costs associated with traditional diagnosis methods. The system's ability to validate input parameters and offer medication reminders further enhances user engagement and promotes better health management.

By offering early detection capabilities and encouraging proactive health monitoring, MediXpert makes a meaningful contribution to improving healthcare accessibility, efficiency, and overall patient outcomes. This project paves the way for future expansions into predicting additional diseases and integrating real-time health data for even more precise and personalized predictions.

ACKNOWLEDGMENT

We express our deepest gratitude to everyone who has contributed to the successful completion of this report on the Medixpert: Multiple Diseases Prediction platform. This endeavor has been made possible through the unwavering support, guidance, and encouragement we have received from several individuals and institutions.

We extend our heartfelt thanks to our principal, Dr. Bhavesh Patel, for his visionary leadership and encouragement, which have greatly inspired us to strive for excellence. The authors also profoundly thank the Head of the Computer Engineering Department, Dr. Vidyullata Devmane, for her constant support and for granting us this valuable opportunity to work on a project of such significance. Her insightful feedback and motivational words have been a guiding light throughout this journey.

We owe special thanks to our guide, Ms. Karuna Borhade, for her exemplary guidance, persistent support, and construc- tive suggestions throughout the development of this project. Her expertise and encouragement have been vital in overcom- ing challenges and achieving milestones, and her commitment to our growth has been truly invaluable.

We are also deeply thankful to all our faculty members, peers, and colleagues in the Computer Engineering Depart- ment for their advice, cooperation, and shared insights, which have enriched the quality of our work. Their inputs have helped shape this project into a meaningful and impactful application.

REFERENCES

[1] A. Dharwal, N. Vyas, V. Sharma, and D. Balla, "Chronic Kidney Disease Prediction Using Robust Approach in Machine Learning," 2023 3rd Int. Conf. on Computational Performance Evaluation (ComPE), 2023.

[2] T. I. Trishna et al., "Detection of Hepatitis (A, B, C and E) Viruses Based on Random Forest, K-nearest and Na¨ıve Bayes Classifier," 2019 10th Int. Conf. on Computing, Communication and Networking Technologies (ICCCNT), Kanpur, India, 2019.

[3] S. S. Nigatu et al., "A Comparative Study on Liver Disease Prediction Using Supervised Learning Algorithms with Hyperparameter Tuning," 2023 Int. Conf. on Advancement in Computation Computer Technolo- gies (InCACCT), 2023.

[4] "Research of Heart Disease Prediction Based on Machine Learning," 2022 5th Int. Conf. on Advanced Electronic Materials, Computers and Software Engineering (AEMCSE), 2022.

[5] M. Giotta et al., "Application of a Decision Tree Model to Predict the Outcome of Non-Intensive Inpatients Hospitalized for COVID-19," Int.
J. Environ. Res. Public Health, vol. 19, no. 20, p. 13016, Oct. 2022.

[6] N. K. Verma, S. Singh, S. S. Chauhan, and D. Kumar, "Disease Prediction with Na¨ıve Bayes Classifier," Vivechan Int. J. Res., vol. 11, no. 1, 2020.

[7] A. Rajkomar et al., "Scalable and Accurate Deep Learning with Elec- tronic Health Records," npj Digital Medicine, vol. 1, no. 18, 2018.

[8] K. Kourou et al., "Machine Learning Applications in Cancer Prognosis and Prediction," Comput. Struct. Biotechnol. J., vol. 13, pp. 8–17, 2015.

[9] C. Sudlow et al., "UK Biobank: An Open Access Resource for Identi- fying the Causes of a Wide Range of Complex Diseases of Middle and Old Age," PLoS Med., vol. 12, no. 3, p. e1001779, Mar. 2015.

[10] A. E. W. Johnson et al., "MIMIC-III, a Freely Accessible Critical Care Database," Sci. Data, vol. 3, p. 160035, May 2016.