ISSN: 2347-6540
International Journal of Latest Research in Engineering and Computing (IJLREC)
Volume 12, Issue 1 : Page No. 24-31
www.ijlrec.com

MULTIPLE DISEASE PREDICTION WEBAPP

[1]Snehashish Pramanick, [2]Shivam Kumar, [3]Ujjwal Kumar,[4]Praveen Kumar, [5]Vandna Bansla
Assistant Professor, vandana.bansla@sce.org.in, Shivalik College of Engineering, Dehradun, India
Student, B. Tech -CSE, Shivalik College of Engineering, Dehradun, India, Email id –snehashish.pramanick04@gmail.coonm
Student, B. Tech -CSE, Shivalik College of Engineering, Dehradun, India, Email id –shivamkradp1411@gmail.com Student,B. Tech -CSE, Shivalik College of Engineering, Dehradun, India, Email id -ujjwalkr7111@gmail.com
Student, B. Tech -CSE, Shivalik College of Engineering, Dehradun, India, Email id -Praveenchoudhary12560@gmail.com

Abstract: Major components of MHM include a sensor -enriched equipment for catching vital indications such as heart rate, blood pressure, oxygen saturation and activity levels. These devices communicate with centralized mobile applications, making users capable of reaching their health data in real time and obtaining personal insights and recommendations. Mobile application also synchronizes data with a safe cloud platform, where advanced analytics algorithm processes the data collected to detect the data to detect indications of changes in health, the data collected to find out. MHMS provides many benefits on traditional health service surveillance approaches. Taking advantage of many sensory types -a more comprehensive approach about a person's health is provided, capable of initial detection of potential health issues and active intervention. In addition, the system promotes the busyness and empowerment of the user through individual reaction and actionable insights, allowing individuals to actively manage their health and well -being.

In summary, the multimodal health monitoring system represents a promising change in healthcare, providing a scalable and cost -effective solution for continuous and personal health monitoring. By integrating weeds

Keywords: DIABETES, HEART, LIVER, KNN, RANDOM FOREST, XGBOOST.

I. INTRODUCTION

Machine learning is a field that focuses on creating predictions by analyzing data from the past. It involves computer systems that study patterns in historical data and experiences to make informed decisions. In recent years, machine learning methods have been increasingly used to detect and identify diseases based on a person's background, medical history, and symptoms. These techniques support healthcare professionals in addressing complex health challenges more effectively.

Many people today prioritize preventive healthcare by taking early measures—such as medication, lifestyle changes, and regular monitoring—rather than waiting for illness to occur. However, predicting the onset of a disease before visible symptoms appear remains a significant challenge. Machine learning can transform healthcare by identifying patterns within large health data sets, allowing for early alerts and preventive strategies.

To ensure reliable disease prediction, it is essential to maintain data security, uphold privacy standards, ensure data quality, and integrate information from various sources such as hospitals, diagnostics, and patient records. A crucial step in this process is feature engineering, which involves removing irrelevant data and refining the remaining information to focus on key health indicators. In this context, techniques such as DESM (Data Extraction and Selection Method) are used to enhance data quality and usability.

In practical terms, machine learning models use labelled datasets—where past cases are already categorized—to predict outcomes like disease progression or risk. These models rely on structured information and apply statistical or deep learning methods to derive results. However, the success of such systems depends on real-world testing.

Clinical effectiveness is measured through patient satisfaction, independent validation studies, and implementation in actual healthcare environments. The goal is to ensure that these tools support medical decision-making without replacing the expertise of healthcare providers. Machine learning, when applied thoughtfully, enhances the ability of doctors, nurses, and public health workers to provide timely and effective care.

Copyright © IJLREC

II. LITERATURE SURVEY

Advancements in information technology have led to an abundance of data in healthcare informatics. Data mining techniques are crucial for extracting meaningful insights from vast amounts of unstructured data. Healthcare data mining holds promise for uncovering hidden trends in the medical field. Data mining is related to artificial intelligence, machine learning, and big data technologies, which involve analyzing, interpreting, and storing large volumes of data. Since the mid-1990s, data mining techniques have been used to explore and identify patterns and links in healthcare data. Healthcare researchers were interested in data mining in the 1990s and early 2000s due to its potential for using predictive algorithms to improve healthcare delivery and model the healthcare system. Data mining employs methods like rule mining, clustering, and classification to analyze data and extract relevant information. Illness diagnosis based on patient data and disease forecasting based on historical data are two well-known applications of data mining in healthcare.

III. DATA PROCESSING

Data Loading and Initial Exploration

To begin working with the dataset, the information is first brought in from a CSV file and organized into a tabular format using a DataFrame, commonly done with the help of the panda's library. Once the data is loaded, simple inspection tools such as

.info(), .describe(), and .value_counts() are used to get a basic understanding of the data. These tools help reveal the types of values in each column, highlight any irregularities, and show how often each category appears in the dataset.

Data Cleaning Process

Cleaning data is a key step in preparing it for any kind of analysis. This process includes scanning for and fixing inconsistencies, eliminating duplicate entries, correcting formatting problems, and handling any incorrect or missing data. Clean data leads to more reliable results and helps prevent issues that might otherwise impact the quality of your findings or predictions.

Dealing with Missing Data

Missing data is a common issue, especially in larger datasets. To check for gaps, methods like. isna(). sum() can be used to count how many values are missing in each column. One way to fill in these blanks is by using backward filling—this means replacing the missing value with the next available one in the same column. This can be done using fillna(method='bfill'). After filling in the missing values, it's good practice to recheck the dataset using. isnull(). sum() to confirm that no gaps remain.

Feature Engineering

Feature engineering is the process of refining and organizing raw data into a usable format that allows machine learning models to detect patterns and make reliable predictions. "The effectiveness of these features has a direct impact on how well a model performs. Properly crafted features can enhance prediction accuracy, speed up model training, and improve how well the model adapts to new, unseen data. On the other hand, poorly prepared features may add unnecessary complexity, introduce irrelevant information, and limit the model's learning capabilities. Key Steps in Feature Engineering: a. Encoding Categorical Data: Categorical variables—such as gender, occupation, or region—must be converted into numerical values, as most machine learning algorithms work with numbers. One common method is label encoding, where categories are assigned numerical codes. For example, gender might be encoded so that 'Female' becomes 0 and 'Male' becomes 1. This conversion helps the model process such data efficiently. b. Standardizing Numeric Features: When a dataset contains numerical features with varying scales—such as age and income—it's important to bring them onto a similar scale. This prevents features with large values from overpowering others during the learning process. A common method for this is standardization, which adjusts values to have a mean of zero and a standard deviation of one, making training more stable and consistent across features.

Model Evaluation and Testing

Final Assessment:

After choosing the most suitable model, its effectiveness is tested on a separate dataset that wasn't used during development. This helps verify that the model performs well when applied to new, unfamiliar data.

Using the Model in Practice:

Copyright © IJLREC

Following successful testing, the model is put to use in real-life situations. It analyzes fresh data to assist with decisions or predictions, while being monitored to ensure it continues to work accurately and reliably.

Feature Engineering

Feature engineering is the process of refining and organizing raw data into a usable format that allows machine learning models to detect patterns and make reliable predictions. “The effectiveness of these features has a direct impact on how well a model performs. Properly crafted features can enhance prediction accuracy, speed up model training, and improve how well the model adapts to new, unseen data. On the other hand, poorly prepared features may add unnecessary complexity, introduce irrelevant information, and limit the model's learning capabilities. Key Steps in Feature Engineering: a. Encoding Categorical Data: Categorical variables—such as gender, occupation, or region—must be converted into numerical values, as most machine learning algorithms work with numbers. One common method is label encoding, where categories are assigned numerical codes. For example, gender might be encoded so that 'Female' becomes 0 and 'Male' becomes 1. This conversion helps the model process such data efficiently. b. Standardizing Numeric Features: When a dataset contains numerical features with varying scales—such as age and income—it's important to bring them onto a similar scale. This prevents features with large values from overpowering others during the learning process. A common method for this is standardization, which adjusts values to have a mean of zero and a standard deviation of one, making training more stable and consistent across features.

Model Evaluation and Testing

Final Assessment:

After choosing the most suitable model, its effectiveness is tested on a separate dataset that wasn’t used during development. This helps verify that the model performs well when applied to new, unfamiliar data.

Using the Model in Practice:

Following successful testing, the model is put to use in real-life situations. It analyzes fresh data to assist with decisions or predictions, while being monitored to ensure it continues to work accurately and reliably.

Model Selection and Training

The process of point design involves relating and organizing meaningful information from raw data to help machine literacy models fete patterns and connections. The quality of these features has a direct impact on how well a model performs. courteously drafted features enhance the model’s capability to make accurate prognostications and draw dependable conclusions from the data. Model Selection Process A critical step in erecting an effective result is choosing the right type of model for the specific problem at hand. Depending on the nature of the task, options may include decision trees, arbitrary timbers, support vector machines, neural networks, and others. Every algorithm has unique advantages and is better suited for specific kinds of data or problem types. Dividing the Data To ensure fair evaluation and avoid bias, the available dataset is resolve into three corridor a training set, a confirmation set, and a test set. The trained data is used to modify the model. The confirmation set helps in fine- tuning the model and opting the most suitable bone. While the test data offers an unbiased evaluation of the model's performance on unfamiliar inputs new, unseen data. assessing Model Performance Once models are trained using the training data, their effectiveness is assessed using the confirmation set. This is done by applying evaluation criteria similar as delicacy, perfection, recall, and the F1 score. Grounded on these measures, adaptations may be made to the model’s parameters to ameliorate its prophetic capability. Choosing the Stylish Model After testing different models and assaying their results, the bone that performs stylish on the confirmation data is named. This final model is also tested on the reticent test set to insure it generalizes well to new data and performs reliably in real- world conditions.

IV. ARCHITECTURE

The current system is designed to identify and predict chronic diseases that are common within particular regions and among specific populations. It uses convolutional neural networks (CNN) along with large datasets to assess the likelihood of disease occurrence. For analyzing this data, several machine learning techniques such as Naïve Bayes, Decision Trees, and K-Nearest Neighbors (KNN) are employed.

This approach has achieved high prediction accuracy—up to 99.69%—making it effective in recognizing patterns linked to chronic illness outbreaks. The system simplifies the application of machine learning to help forecast diseases in communities where such health issues frequently occur. To ensure reliability, the models have been tested using real hospital records collected from healthcare centers in central China. A system called CNN-MDRP (Convolutional Neural Network–based Multimodal Disease Risk Prediction) has been introduced, which handles both organized data (like medical records and reports) and unstructured data (such as doctors’ notes).

Copyright © IJLREC

The framework supports disease prediction by allowing structured health information to be entered directly into the model. Patients or healthcare professionals can input symptoms, which are then used to train the system to predict potential conditions. Different algorithms work together to improve prediction quality—KNN helps in classifying symptom data, while Naïve Bayes is used to predict diseases based on those symptoms. Logistic regression is also applied to identify key features by analyzing outcomes from decision tree models.

Accuracy-:

$$\frac{TruePositive+TrueNegative}{TruePositive+TrueNegative+FalsePositive+FalseNegative}$$

$$\text{Precision} = \frac{TruePositive}{TruePositive+FalsePositive}$$

$$\text{Recall} = \frac{TruePositive}{TruePositive+FalseNegative}$$

$$\text{F1-Measure} = \frac{2\times precision\times recall}{precision+recall}$$

Fig. 1

I. ALGORITHM TECHNIQUES

KNN

K-Nearest Neighbors (KNN) is a straightforward and flexible machine learning technique that is commonly used for both classification and regression tasks. In the context of healthcare, it serves as a useful tool for predicting the presence of diseases based on patient symptoms. By comparing new cases to previously known data, KNN helps identify the most likely diagnosis by analyzing the similarity between data points.

The algorithm works by finding the "k" closest data points—known as neighbors—to the new input and assigning a category based on the majority class among them. The value of "k" is chosen by the user, but its optimal value often depends on the nature of the data. Using a larger "k" can help minimize the effect of outliers or noise in the dataset.

To determine how close different data points are, KNN uses distance metrics. For categorical features, Hamming distance is commonly used, while continuous numerical features often require normalization to bring values into a comparable range, usually between 0 and 1. This becomes especially important when dealing with a mix of numerical and categorical data, as inconsistent scaling can impact the model's accuracy.

When a new symptom or feature is entered, the algorithm measures its distance from existing data and identifies the closest matching group, helping healthcare providers make more informed predictions about potential illnesses.

Decision tree

A decision tree is a framework that divides a large collection of records into smaller sets using simple decision trees. Generated sets become more similar with each division. A decision tree model is a collection of rules that divide a heterogeneous population into smaller, homogeneous groups concerning a specific aim. The target variable is typically categorical, and the decision tree is used to determine the likelihood of a record falling into each category or to classify the record by assigning it to the most likely class.

Copyright © IJLREC

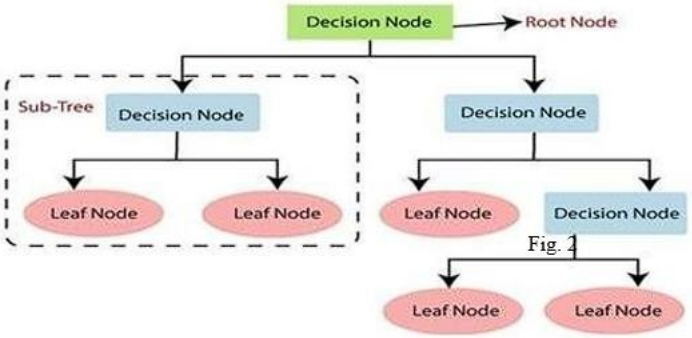

Fig. 2

Random Forest

Random Forest is a powerful machine learning approach that works by creating several decision trees using randomly selected parts of the training data. Instead of relying on a single tree, it merges the outcomes of all trees to make a final prediction, which improves reliability and reduces the risk of overfitting. This technique performs well with datasets that contain a mix of numerical and categorical variables, making it particularly useful in medical scenarios like liver disease diagnosis. Its ability to manage high-dimensional data and discover intricate relationships among features makes it a strong choice for complex health data. One of the key advantages of Random Forest is its ability to rank the importance of input features. In medical analysis, this allows doctors and researchers to pinpoint which patient details contribute most to disease prediction, offering valuable insights that can support better diagnosis and treatment strategies.

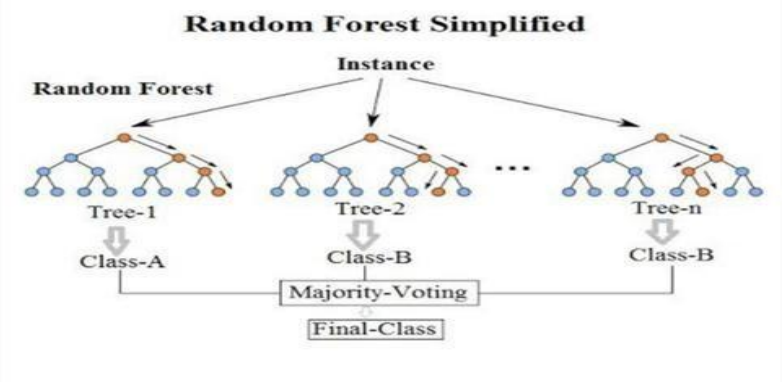

Fig. 3

Support Vector Machine (SVM)

Support Vector Machine (SVM) is a supervised learning method used for classification tasks. It works by identifying the optimal boundary—called a hyperplane—that best separates data points belonging to different categories in a given feature space. This makes it effective for handling both linear and complex, nonlinear classification problems.

SVM is especially beneficial when working with datasets that have many features or limited sample sizes. Its design helps reduce the chances of overfitting, making it reliable even with noisy or sparse data. To capture more complex patterns, SVM uses different kernel functions such as linear, polynomial, and radial basis functions (RBF), which help transform data into higher dimensions where it becomes easier to separate.

One of the key strengths of SVM is its focus on maximizing the margin between classes, which leads to better generalization on unseen data. This makes SVM a valuable tool in clinical environments, where both predictive accuracy and clarity of results are essential for decision-making.

Copyright © IJLREC

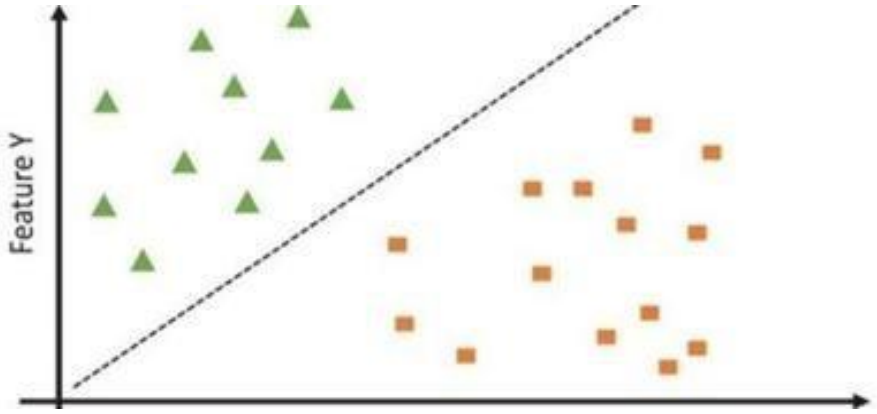

Fig. 4

XG Boost Algorithm

XG Boost algorithm steps:

Step 1: Initialize with a Base Tree

Start by building a tree with just one leaf. This serves as the initial model.

Step 2: Make Initial Predictions and Calculate Errors

For the first iteration, use the average of the target values as your initial prediction. Then, compute the difference between these predictions and actual values using a suitable loss function (such as squared error). In the following iterations, calculate the residuals based on predictions made by the previous trees.

Step 3: Evaluate Similarity Score

Compute the similarity score for splitting nodes using the formula:Similarity Score = (Sum of Gradients) ² / (Sum of Hessians + λ)

Here, the gradient represents how far predictions are from the actual target, the Hessian refers to the second derivative or curvature, and λ is a regularization term that helps control model complexity.

$$Similarity\ Score = Gradient \frac{Gradient^2}{Hessian + \lambda}$$

Step 4: Choose the Best Node to Split

Use the similarity score to identify the best node for splitting. A higher score indicates that the group of data points in the node is more consistent, which is desirable.

Step 5: Compute Information Gain

Determine the improvement gained from a split by calculating the difference between the similarity score before and after the split. This shows how much more uniform the data becomes due to the split.

$$Information\ Gain = Left\ Similarity + Right\ Similarity - Similarity\ for\ Roots$$

Step 6: Build the Full Tree

Continue splitting nodes using the similarity and gain measures until the desired depth or size of the tree is reached. Use

Copyright © IJLREC

regularization parameters to avoid overfitting and to trim unnecessary branches (pruning).

Step 7: Make Predictions Using the Tree

Use the new tree to predict the residuals (errors) of the model, which will be used to adjust the overall predictions.

Step 8: Update Residuals

Adjust the residuals with the formula:

New Residual = Previous Residual - (Learning Rate × Prediction)

The learning rate controls how much influence each new tree has on the final model.

Step 9: Repeat for Additional Trees
Repeat the entire process to build more trees, each time updating the residuals and improving the model's prediction accuracy

$$New\ Residuals = Old\ Residulas + \rho \sum Predicted\ Residuals$$

V. CONCLUSION

The creation and practical use of a multimodal health prediction system mark an important step forward in healthcare innovation. By bringing together information from various sources—such as fitness trackers, genetic profiles, clinical records, and patient feedback—this system provides a fuller and more accurate picture of an individual's health.

Through careful analysis of these different types of data, healthcare professionals are better equipped to detect early signs of illness, recognize risk patterns, and make informed decisions tailored to each patient. This approach not only improves the accuracy of predictions but also allows for timely interventions and more effective treatment strategies.

What sets this system apart is its ability to evolve as more data becomes available. With each new set of patient information, the system can refine its insights, ensuring that health evaluations and recommendations stay up to date and relevant. This ongoing refinement supports healthcare workers in diagnosing and managing conditions more effectively.

In summary, a multimodal health prediction system holds great promise in improving the way healthcare is delivered. It supports early diagnosis, helps make better use of resources, and can lead to better long-term outcomes for patients. As with any new healthcare tool, ongoing research and real-world testing are needed to enhance its reliability and address any remaining challenges.

VI. FUTURE SCOPE

The evolution of healthcare monitoring systems is being driven by the ongoing advancement and application of machine learning (ML) and deep learning (DL) technologies. As these technologies continue to mature, several areas stand out as crucial for future exploration and development:

[1] Personalized Treatment Approaches

Future healthcare systems will increasingly focus on tailoring medical treatments to individual patients. By analyzing genetic information, daily habits, and environmental exposures, ML and DL models can help create care plans that are better suited to each person's unique needs.

[2] Continuous and Real-Time Data Analysis

Efforts are underway to create systems that can monitor patient data in real-time, drawing from sources such as wearable devices, medical sensors, and health records. These tools can provide timely warnings and support quicker medical responses.

[3] Advanced Health Predictions

By improving prediction models, healthcare professionals will be better able to forecast disease progression, hospital readmissions, and the demand for medical services. These insights will allow for early intervention and more efficient use of resources.

[4] Integration of Diverse Health Data

A major challenge in healthcare is bringing together different types of medical information—from lab reports and imaging results to genomic data and patient-reported symptoms. Improved systems are needed to link and analyze this data in a meaningful and unified way.

Copyright © IJLREC

[5] Ethical and Privacy Safeguards

As digital health systems handle increasingly sensitive data, maintaining patient privacy and ensuring ethical use is critical. Researchers and healthcare providers must implement strong safeguards, secure data-sharing practices, and obtain informed patient consent.

[6] Transparent and Trustworthy Models

There is a growing need for ML and DL tools that are not only accurate but also explainable. Healthcare providers must be able to understand how a model reached its conclusions so they can confidently apply it in clinical decision-making.

[7] Expanding Remote Care Access

Enhancing remote monitoring and telemedicine technologies can bridge gaps in healthcare delivery. These innovations have the potential to serve patients in rural or underserved areas, improving overall access and reducing inequalities in care.

[8] Collaborative Studies and Clinical Validation

Large-scale, collaborative research efforts involving clinicians, scientists, and regulators will be essential. Conducting clinicaltrials and real-world evaluations will help assess the effectiveness, safety, and reliability of AI-driven healthcare tools before they are widely adopted.

REFERENCES

1. Mahajan et al. (2023) examined how ensemble learning methods—such as bagging, boosting, stacking, and voting—can be applied to predict diseases like diabetes, skin conditions, kidney disorders, liver problems, and heart ailments. The review emphasized the advantages of combining multiple algorithms to improve diagnostic accuracy.
2. Chauhan, Patel, and Shah (2021) explored a range of machine learning models aimed at predicting multiple diseases using symptoms as primary input data. Their work highlighted how these algorithms could support clinical assessments and early diagnoses.
3. Gaurav, Malviya, and Tripathi (2023) focused on identifying early signs of heart disease by applying feature selection techniques such as Chi-Square, ANOVA, and Mutual Information. Their findings showed that choosing the right input features significantly boosts prediction accuracy.
4. Yang et al. (2023) proposed a hybrid approach that combines deep learning with machine learning to detect cardiovascular conditions. Their model used advanced feature selection and model-stacking strategies, leveraging a large and diverse dataset.
5. Patel and Sharma (2023) assessed several machine learning classifiers to determine heart disease risk. The study highlighted that selecting relevant features was a critical factor in achieving high accuracy in risk predictions.
6. Singh and Singh (2022) developed a system capable of predicting multiple diseases using an array of machine learning techniques. Their work aimed to enhance diagnostic tools and support healthcare professionals in decision-making.
7. Khalil and Jones (2022) introduced a framework for diabetes prediction that uses ensemble multi-classifier models. Their system tackled the challenge of class imbalance and demonstrated improved accuracy in detecting diabetes.
8. Appelbaum and Rinard (2024) introduced a model named Prism, developed for assessing pancreatic cancer risk. It was trained and validated on a vast dataset from clinical records across the United States, aiming for real-world applicability.

Copyright © IJLREC