Journal of Neonatal Surgery
ISSN(Online): 2226-0439
Vol. 14, Issue 32s (2025)
https://www.jneonatalsurg.com

OPEN ACCESS

Enhancing Healthcare Outcomes Through Big Data Analytics Using Advanced Data Mining and Classification Techniques

Sreevidya N R[1], Dr. L. Sudha[2]

[1]Research Scholar, Department of Computer Science, VET Institute of Arts and Science College,Erode, TamilNadu, India

[2]Associate Professor, Department of Computer Science, VET Institute of Arts and Science College, Erode, TamilNadu, India

Cite this paper as: Sreevidya N R, Dr. L. Sudha, (2025) Enhancing Healthcare Outcomes Through Big Data Analytics Using Advanced Data Mining and Classification Techniques. *Journal of Neonatal Surgery*, 14 (32s), 9012-9027.

ABSTRACT

Big data analytics is transforming the healthcare industry by enabling deeper insights through data mining and classification techniques. These technologies improve decision-making, diagnosis, and overall healthcare outcomes by analyzing vast volumes of medical data. However, existing methods often struggle with the accuracy, scalability, and handling of high-dimensional healthcare datasets, leading to limited predictive performance and inefficient clinical interventions. To address these challenges, this study proposes a framework called Random Forest Classification-based Predicting Patient Readmission Risk (RFC-PPRR). This framework leverages the ensemble learning capability of Random Forest to effectively process large electronic health record (EHR) datasets and accurately predict patient readmission risk within 30 days of discharge. The proposed method enables hospitals to identify high-risk patients early, facilitating the implementation of preventive measures and more effective resource allocation. An experimental evaluation of real-world healthcare datasets demonstrated improved prediction accuracy, reduced false positives, and enhanced model interpretability compared to traditional classification methods. These outcomes suggest that RFC-PPRR can significantly contribute to reducing avoidable readmissions and optimizing patient care strategies.

Keywords: Big Data Analytics, Healthcare, Random Forest, Patient Readmission, Classification, Predictive Modeling.

1. INTRODUCTION

Big data analytics has enabled the healthcare industry to save money, manage resources more efficiently, and deliver better care to patients. Healthcare professionals could make better treatment decisions if they could uncover useful patterns and information in large amounts of electronic health data[1]. Big data analytics helps address many problems, but one of the most important ones is determining the likelihood of a patient needing to return to the hospital, particularly within 30 days of discharge. It is a crucial way to determine the quality and effectiveness of healthcare. It will quickly identify those at high risk and reduce the burden on healthcare systems by utilizing accurate prediction algorithms[2].

Healthcare systems worldwide are concerned about 30-day hospital readmissions, as they are detrimental to patients' health and costly. Medicare spends more than $17 billion a year in the US on unexpected readmissions[3]. People frequently had to return to the hospital because they didn't receive the correct treatment after leaving, failed to follow the doctor's instructions, or became suddenly ill. That is why it is crucial to have reliable models to support predictions. However, the numerous distinct aspects of healthcare data can make it challenging for traditional statistical methods, such as logistic regression, to estimate the likelihood of readmission accurately. Healthcare personnel need to utilize the most advanced machine learning algorithms to accurately predict the likelihood of readmissions, enabling them to take proactive measures and improve patient outcomes[4-5].

Although machine learning is becoming increasingly common in healthcare, accurately predicting which patients will require readmission remains a challenge. Many of the models currently employed can't work with datasets that have a large number of dimensions because they overfit and perform poorly on new data[6]. One significant reason why more advanced models aren't used in hospitals is that they are difficult to understand. When the datasets are unbalanced, meaning more events lead to readmission than those that don't, it is also very challenging to train and evaluate models. One needs models that can easily manage complex healthcare data, are easy to understand and can adapt to meet future demands, thereby overcoming these limitations and making accurate predictions about readmission risk [7-8].

.

It utilizes the RFC-PPRR framework to help us determine how to address these problems. One can employ random forest classification to assess the likelihood of a patient being readmitted. Random Forest is a type of ensemble learning that is very strong, works well with enormous datasets, and performs exceptionally well with data that has a high dimensionality. Random Forest makes forecasts that are more accurate and less likely to overfit by using the results of several decision trees. Model interpretability is a crucial element in making treatment decisions, and measuring the significance of each characteristic is helpful in this regard. The RFC-PPRR system can manage a large amount of EHR data and provide more accurate and easier-to-understand estimates of the risk of readmission within 30 days by utilizing these features. The framework's purpose is to help identify high-risk patients early, simplify the provision of treatments, and use resources effectively to reduce readmissions that could have been avoided.

Contribution of this paper,

This paper proposes the RFC-PPRR framework, which combines Random Forest classification with healthcare data to accurately predict patient readmission risks, offering an innovative approach to healthcare data mining.

The RFC-PPRR framework outperforms traditional methods by significantly enhancing prediction accuracy and reducing false positives, resulting in more reliable patient risk stratification and improved clinical decision-making.

By predicting readmission risks, RFC-PPRR enables healthcare providers to proactively implement preventive measures, improve resource management, reduce readmissions, and ultimately enhance patient outcomes and care efficiency.

2. RELATED SURVEY:

A significant amount of research has been conducted over the past several years on how machine learning (ML) can enhance the accuracy of predictions about hospital readmissions and clinical decisions. This study presents a summary of the most important research conducted, focusing on its methodologies, aims, and limitations.

Kalusivalingam et al. (2025) [9] studied how well the Random Forest and Gradient Boosting algorithms could predict how frequently patients would have to go back to the hospital. Their models outperformed ordinary logistic regression, demonstrating the effectiveness of ensemble methods in handling complex healthcare data. The research didn't focus sufficiently on how easily the models can be grasped, which is highly significant for their use in clinical settings.

Chen et al. (2025) [10] employed several machine learning models, including XGBoost, to predict the number of people with acute pancreatitis who would need to return to the critical care unit. They employed feature selection methods, such as LASSO and RFECV, to enhance the model's performance. The research can not have been particularly effective for a wider group of individuals, as the sample size was so small.

In 2022, Michailidis et al.[11] employed several techniques, including Support Vector Machines and decision trees, to construct machine learning models that could predict when patients would need to return to the hospital. Even though their plan seemed promising for predicting readmissions using machine learning, they didn't test how well the models performed on all types of patients.

Liu (2025) [12] examined various machine-learning models to determine whether they could predict the likelihood of hospital readmissions within 30 days among individuals with diabetes. The research revealed that different models performed better or worse on critical criteria. This study only examined individuals with diabetes; therefore, the findings can not apply to everyone.

Miswan et al. (2021) [13] used a multitude of preprocessing methods on ML classifiers to find out which patients would need to go back to the hospital. The model improved with LASSO and other techniques for feature selection. Unfortunately, the research didn't examine how these approaches can be applied with ensemble learning methods.

Khalid et al. (2022) [14] utilized health insurance data to teach Long Short-Term Memory (LSTM) networks how to guess how frequently patients would need to go back to the hospital. Their superior performance compared to standard classifiers revealed that temporal data can be exploited for prediction. But insurance data don't always include all the details of a patient's care, which might make it impossible to make reliable forecasts.

Chen et al. (2025) [15] employed machine learning (ML) models, such as LightGBM, to identify patients who are likely to require readmission to the critical care unit following an intracerebral hemorrhage. Their findings were impressive; however, they only examined one condition, so their conclusions cannot be applied to other medical contexts.

Wang and Zhu (2024) [16] used a Random Forest technique that included Natural Language Processing (NLP) and other ways to eliminate unfairness in predicting readmissions after joint replacement procedures. Their technique performed effectively in addressing biases in predictions, but it will only apply to one type of operation.

Tang et al. (2022) [17] developed a multimodal spatiotemporal graph neural network that could predict all-cause hospital readmissions within 30 days. Their model performed better than the others, as it utilized multiple types of data. However, since graph neural networks are so complicated, they will not be as helpful for treatment.

The Random Forest Classification-based Predicting Patient Readmission Risk (RFC-PPRR) technique utilizes a novel approach to analyze a large amount of electronic health record data to determine the overall 30-day readmission risk for distinct patient groups. RFC-PPRR finds a balance between accuracy, scalability, and model transparency by employing the ensemble learning properties of Random Forest. This makes it simpler to utilize in more clinical contexts, as earlier research has either focused on specific scenarios or employed complex models that were difficult to comprehend.

3. PROPOSED METHOD:

This work provides information on the RFC-PPRR framework, which can be used in conjunction with Random Forest Classification to estimate the likelihood of patients being readmitted to the hospital within the next 30 days. The objective is to address the most pressing issue in this region, where the cost of healthcare systems increases whenever patients are readmitted to the facility. It can indicate that the therapy was not effective or that the discharge plan is insufficient. Through accurate forecasting of the likelihood of readmission, along with the implementation of measures to prevent it, it will be possible to achieve better outcomes for patients and utilize resources more effectively. The RFC-PPRR architecture achieves this by combining a large number of decision trees into a single, highly accurate classifier through the use of the Random Forest technique. The usage of ensemble learning is shown here. Random Forest is superior to traditional single-tree models in terms of generalization and preventing overfitting. Random Forest incorporates the predictions of a large number of trees that were initially generated independently. Given that healthcare data frequently involves complex, nonlinear relationships and noisy characteristics, the ensemble technique is well-suited for handling this type of data.

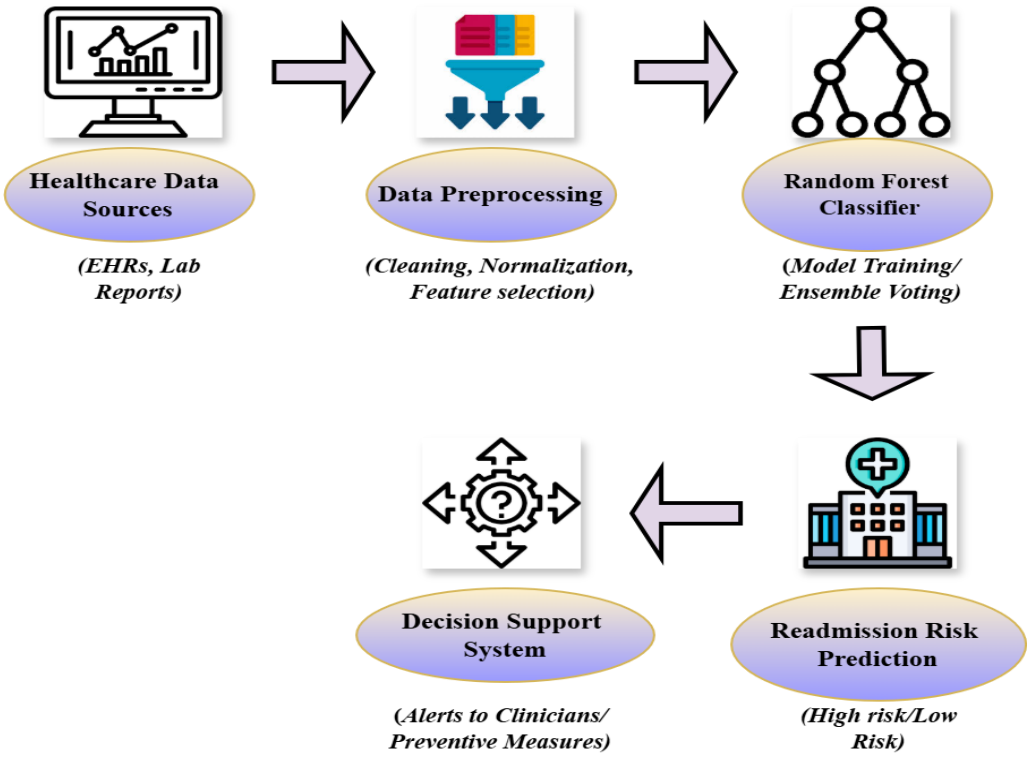

Figure 1: Architecture of proposed method

Through the use of clinical criteria, patient demographics, and data from previous hospitalizations, RFC-PPRR can provide patients with an accurate risk assessment within 30 days after their discharge from the hospital. By identifying high-risk patients at an early stage and taking appropriate action, healthcare professionals can improve the overall quality of treatment and reduce the number of patients who require avoidable readmissions. Since the framework is capable of making predictions, this is feasible. The RFC-PPRR system (Figure 1) comprises components responsible for data processing, feature selection, training the Random Forest model, generating predictions, and conducting accurate evaluation.

The RFC-PPRR framework is a large machine-learning pipeline that utilizes a substantial amount of high-dimensional electronic health record (EHR) data to predict which patients are likely to require a return visit to the hospital within the next thirty days. As a result of the framework's architecture, scalability, interpretability, and high prediction accuracy are the most crucial aspects it must address to tackle significant challenges in healthcare analytics. There are five primary phases involved in this technique. The first step is to gather information from electronic health records (EHRs) that include a significant

number of clinical features and then prepare it. Following that, these datasets are cleaned, encoded, standardized, and balanced to make them suitable for use in model construction. The next step is called Feature Selection and Transformation, which employs statistical and machine learning techniques to reduce the number of dimensions, improve the model's efficiency and ability to generalize and identify the variables that are most important within the large dataset. In the third stage, referred to as "Random Forest Model Construction," a collection of decision trees is developed to precisely and reliably categorize patients according to the likelihood of their readmission to the hospital. Lastly, the Prediction and Decision Support stage enables healthcare professionals to make accurate assumptions that aid in identifying patients at high risk and developing individualized discharge plans for each patient. When making healthcare decisions, it is essential to have these procedures in place before utilizing data-driven intelligence.

Data Collection and Preprocessing:

The RFC-PPRR method is based on collecting accurate data and carefully organizing EHR information, as shown in Figure 2. During the data collection phase, a large amount of data is collected from the systems of healthcare organizations. There are several types of patient data in these databases. Some examples of demographic data include age, gender, and race. Other helpful indications include summaries of admissions and discharges, clinical procedures, lab test results, diagnostic codes (such as ICD-10), and information about previous readmissions. It will utilize this large and diverse dataset to train and test algorithms that make predictions.

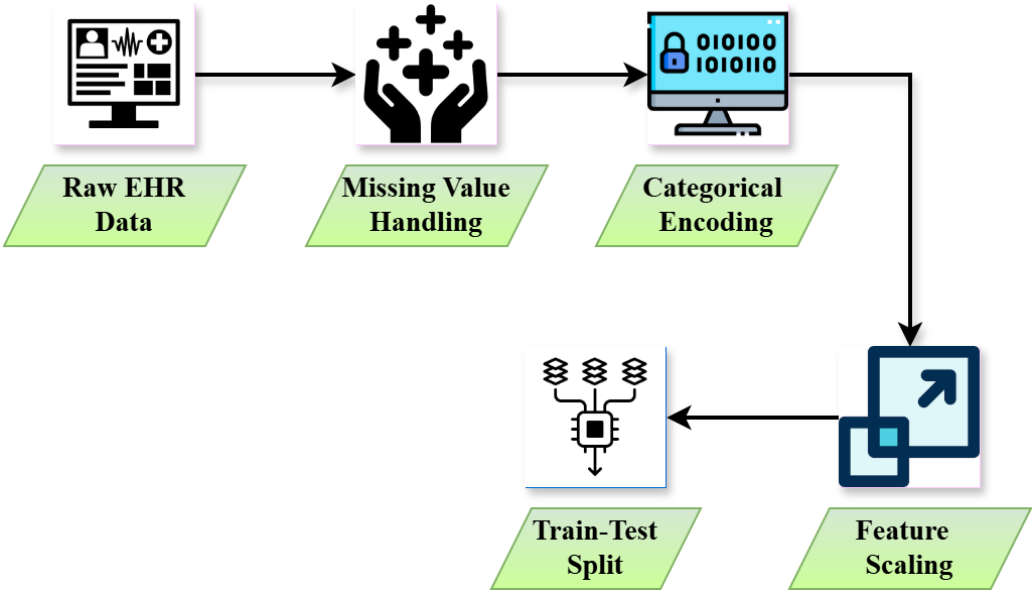

Figure 2: Data preprocessing pipeline

3.1.1 Raw EHR Data:

The first step is to get the raw, unedited records from hospital databases. These are known as EHR Data. This information typically includes various details, such as the patient's age and gender, the dates of admission and discharge, the codes for procedures (both diagnostic and procedural), test results, prescription information, and a history of readmissions. Due to their vastness, diversity, and potential for inaccuracies or missing information, these documents require careful preparation.

3.1.2 Missing Value Handling:

The next stage is to handle missing values, which are prevalent in medical data because of irregular updates, missed tests, or system constraints. This stage makes sure that the data is complete by employing several methods to fill in the gaps. For simple datasets, the mean, median, and mode are sufficient. But for more complicated datasets, you may require more sophisticated methods like K-Nearest Neighbors (KNN) imputation or multivariate imputation. By handling missing values correctly, the machine learning algorithm may find important patterns even when data is absent and prevent model bias.

The characteristic n doesn't have any numerical qualities; hence, in (1), the mean is employed instead of n_x.

$$n_x = \frac{1}{i}\sum_{y=1}^{i} n_y \quad (1)$$

3.1.3 Encoding by Role:

Categorical Encoding is used to convert non-numerical categorical data, such as gender, diagnostic code, or hospital unit, into numbers during imputation. This translation is crucial since machine learning models, such as Random Forests, require numerical input. Label encoding may be used to encode ordinal properties, whereas one-hot Encoding is commonly used to

make binary columns for each category. This step is the last one for keeping data for future algorithmic use.

Scaling the Features:

After the dataset has been numerically encoded, Feature Scaling is used to standardize the range of the independent variables. Random Forests can work with data that hasn't been scaled, but when they are combined with other models or hybrid architectures, scaling could help them work better and make them easier to compare. Min-max scaling and similar approaches scale the size of features to fit within a certain range, typically [0, 1]. Standardization, on the other hand, changes the size of characteristics such that their mean is 0 and their standard deviation is 1. This maintains training stability and ensures that features remain useful.

a) Min-Max Normalization involves scaling the size of features to a set range of values between 0 and 1 (2).

$$n'_x = \frac{n_x - \min(n)}{\max(n) - \min(n)} \quad (2)$$

b) Standardizing the Z-Score: This method centers the data and makes the variance the same:

$$n'_x = \frac{n_x - \mu}{\sigma} \quad (3)$$

Here, (3) explain what the mean (μ) and standard deviation (σ) of the feature are.

3.1.5 How to Split the Train and Test:

Lastly, the Train-Test Split means splitting the cleaned and changed dataset in two. This allows you to train the model on half of the data and test its performance on the other half. It is common to split data into two groups: training and testing. In this manner, the model may learn from the first one while still having enough data to see how well it works in general. This distinction is crucial for ensuring the model doesn't overfit, and for testing it on data it hasn't seen before.

Finally, a separate system for testing and training.

For example, dataset G has i samples in total. The split ratio $f \in (0, 1)$ divides the two halves.

Training Set: $G_{\text{train}} = \lfloor f.i \rfloor$

Test Set: $G_{\text{test}} = i - \lfloor f.i \rfloor$

The experimental design decides whether the normal value of f=0.8 or f=0.7.

This pipeline begins with the acquisition of raw EHR data and then proceeds to prepare the data in a structured format. This sets the stage for reliable and clear predictive modeling. It can ensure that the data is of high quality, accurate, and compatible with machine learning methods like the RFC-PPRR by carefully preparing ahead. This will enable us to accurately predict readmissions within 30 days and provide us with valuable clinical information.

Feature Selection and Transformation:

Machine learning algorithms will identify numerous variables in EHRs and other forms of high-dimensional healthcare data that are unnecessary, repetitive, or noisy. The RFC-PPRR architecture prevents models from being trained on features that aren't needed or don't matter by employing feature selection and transformation. It conducts a correlation analysis to determine whether two qualities are statistically related. Redundant characteristics provide the same information and are highly linearly correlated. For instance, the Pearson correlation coefficient is higher than 0.9. It can be possible to reduce dimensionality without losing our ability to make predictions if one removes one of the connected variables.

The Random Forest classifier employs Recursive Feature Elimination (RFE), a wrapper-based method that the system uses to reduce the number of features further. RFE's purpose is to find the optimal subset by repeatedly removing features that aren't important to the model. It does this by examining the importance of each trait according to the model's estimates. A popular way to measure these grades is to use Gini impurity or mean loss in accuracy. In that way, the traits that are kept will still be valuable when it's time to make predictions.

Random Forest Algorithm:

RFC-PPRR utilizes the Random Forest method, as shown in Figure 3, which is a powerful technique for learning from a large number of samples. The algorithm appears to function well with healthcare data, which can be quite complex and challenging to predict. It can handle high-dimensional feature fields, is noise-tolerant, and can model connections that aren't straight lines; thus, it's a strong candidate for predicting 30-day patient readmission. A complicated web of clinical, demographic, and behavioral variables shapes this disease. Traditional single-lever decision trees tend to overfit and don't generalize well enough. Random Forest, on the other hand, harnesses the pooled knowledge of many learners to produce better predictions.

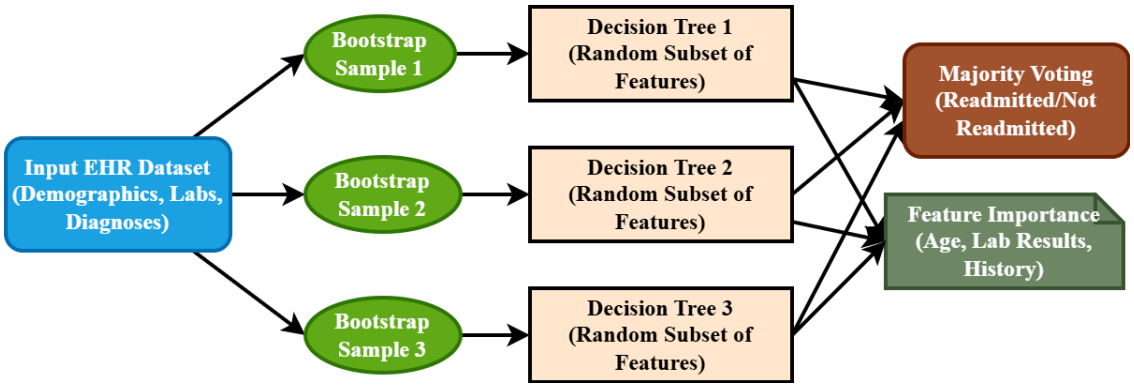

Figure 3: Random Forest model construction for classification

Bootstrapping to create different training subsets:

Building a Random Forest model cannot begin until bootstrap samples have been generated. The training data is divided into several random groups, and each decision tree in the forest is trained in one of these groups. It can easily create these subgroups by modifying certain variables in the training set. This means that certain data points will appear more than once in a sample, while others will not appear at all. This bootstrapping strategy makes the trees more statistically diverse, which might make it less likely that they would fit too closely to particular patterns in the original dataset. This unpredictability makes the model more useful for a wide variety of patient profiles, which is particularly beneficial in healthcare records that often contain unique data combinations that overlap.

Feature subsampling:

It could help minimize the chances of overfitting and correlation. Random Forest is a machine-learning method for training decision trees. It achieves this by selecting a random subset of characteristics at each node. It doesn't utilize all of the traits to discover the best split; instead, it uses a randomly picked group of them. Feature subsampling is a strategy that reduces the connection between trees, making processing faster. This strategy prevents a small number of strong features from dominating the model in high-dimensional electronic health record (EHR) data, where many items will be closely related (such as lab results or diagnostic codes). This makes the ensemble more diversified and stronger.

Ensemble voting to aggregate predictions:

A majority vote is used to integrate the output of each trained decision tree. When it comes to categorizing anything, such as deciding whether or not to readmit a patient, each tree provides a positive or negative answer. It will make the final forecast by examining the class that the majority of the trees indicate. This method of ensemble voting yields far more accurate and dependable predictions, as it reduces the likelihood that individual trees will make inaccurate decisions due to bias or noise in their training subsets.

The feature significance ranking element of Random Forest is very useful in medicine. Another nice thing about it is that it can make quite accurate predictions. The technique examines how each feature affects the total decrease in impurity across all forest splits during training. Researchers and doctors will gain a deeper understanding of the factors that have the most significant impact on the model's predictions by examining these relevance ratings. Age, previous readmissions, other health issues, and test results are some of these considerations. This information is not only easy to read, which makes things clearer, but it also provides useful insights into patient risk factors, aiding in the planning of care and actions after discharge.

Classifier for Fixing Problems in Electronic Health Records:

The RFC-PPRR design of the Random Forest architecture covers all the critical aspects required for accurate healthcare prediction. Some of these attributes include being accurate, resilient, and easy to understand. To address the complex, noisy, and diverse nature of EHR data, the model employs bootstrapped sampling, random feature selection, ensemble learning, and built-in interpretability, as illustrated in Figure 4. Thus, it serves as a high-performance classifier and a tool to aid doctors in making informed decisions that enhance their judgment and reduce unnecessary hospital readmissions. There is both theoretical and real-world data to support the Random Forest model generation process used by the RFC-PPRR framework. A model is developed to predict patient readmissions within 30 days by utilizing techniques such as bagging, feature randomization, majority aggregation, and low-correlation weak learners. The model is quite realistic, can be applied in many different circumstances, and is straightforward to understand. It will be particularly useful for clinical decision support systems, as it aligns well with the characteristics of actual healthcare data.

Features-based analysis:

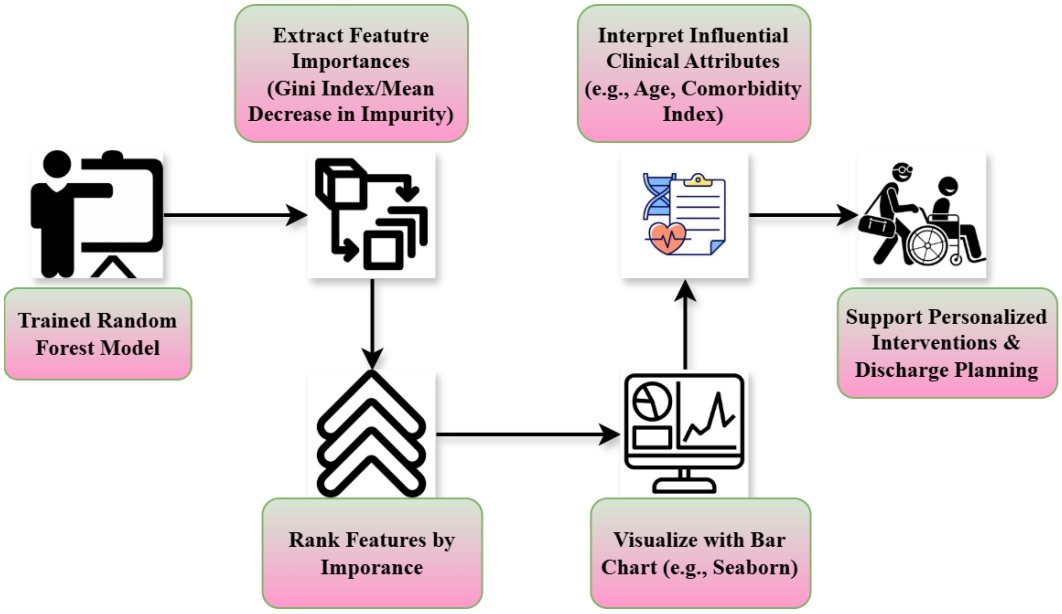

Figure 4: Feature importance analysis

It is highly crucial to evaluate the Random Forest model after training it in the RFC-PPRR framework using EHR data to get useful clinical insights. The first thing you need to do is use the built-in tools in Random Forest to find the bits that illustrate how essential each trait is. The Gini Index and the Mean Decrease in Impurity (MDI) are two prominent techniques used to determine the importance of these levels. The Gini Index could show how well a characteristic divides groups at each node of a decision tree. As the Gini impurity decreases, the feature in question becomes more relevant. The MDI can help us acquire this information from all the trees. Then, we can figure out which traits always make it harder to categorize.

After the significance scores have been figured out, the model puts the qualities in order from most essential to least important, depending on how much weight each one has in the model's decision-making process. A patient's comorbidities, the duration of their condition, abnormal lab results, and their history of readmission are among the most critical clinical factors that impact the probability of readmission. It can assess the real-world benefits of these quantitative traits by applying them in a therapeutic setting. Patients who have a high relevance score for "previous readmission within 30 days" may require extra support organizing their care since they are always unstable.

A bar chart indicates the order of feature significance, making it easier to understand and communicate with stakeholders. This chart makes it simple for everyone, from doctors to hospital managers to data scientists, to see which variables have the most impact on the model's predictions. The length of each bar tells one how essential that feature is for reducing model error. These findings may make the AI model more transparent and trustworthy, and they could also provide caregivers with valuable information.

The primary purpose of this interpretability pipeline is to support planning for release and individualized care. Healthcare providers can tailor patient education, medication modifications, home health service suggestions, and follow-up therapy to meet the needs of each patient by identifying which patients are most likely to be readmitted. The RFC-PPRR approach can always identify risks, but it also has two additional benefits: it helps patients recover and prevents them from needing to return to the hospital. This is possible when the model's explainability and clinical actionability are put together.

As evaluated by the Gini Index, treatment lowers the levels of impurity.

Gini:

The Gini importance, which is the average reduction in impurity, is used by Random Forests to find out how significant each feature is for making judgments. To minimize impurity, decision trees attempt to separate data based on a feature. The Gini index is one way to quantify class mixing. The Gini coefficient examines all the characteristics in the forest and rates them based on how much they contribute to reducing pollutant levels. A feature that consistently leads to purer splits receives a higher relevance score, as it decreases impurity by a larger amount. This strategy is useful for handling complex healthcare data, as it is designed to identify nonlinear and hierarchical relationships within the data. "Length of stay" is a very important indicator for predicting readmissions since it often shows up in early splits.

Permutation:

Another technique for assessing the ease of understanding a model is to examine its permutation significance. It shows how each feature genuinely changes the model's projected performance. This method allows us to modify the value of a feature randomly across all samples while keeping all other data points unchanged. This is what makes it different from the outcome. The decline in model accuracy, as indicated by the drop in the AUC or F1-score, highlights the significance of the feature. This method doesn't predict how much a feature helps lower impurity; instead, it gives a direct assessment of how well it generalizes. Permutation relevance is most effective in clinical contexts because it demonstrates tangible consequences. For instance, if "lab abnormalities" are changed and performance drops significantly, this suggests that it is extremely useful for forecasting readmissions.

Health advantages and the capacity to focus on key details:

Bar graphs and sorted lists are two types of visualizations of feature relevance that help people comprehend Random Forests better. These pictures give doctors more influence and help people understand why they make certain decisions. "Number of prior hospitalizations," "discharge disposition," and "comorbidity score" are three of the most significant things that different models use to attempt to predict hospital readmissions. By identifying crucial risk factors, doctors and nurses can design treatment plans tailored to each patient, respond promptly to emergencies, and empower patients to make informed decisions. Responsibility, ethics, and trust are particularly crucial in therapy settings. This implies that the choices of models must be transparent and make sense.

Decision support in predicting outcomes:

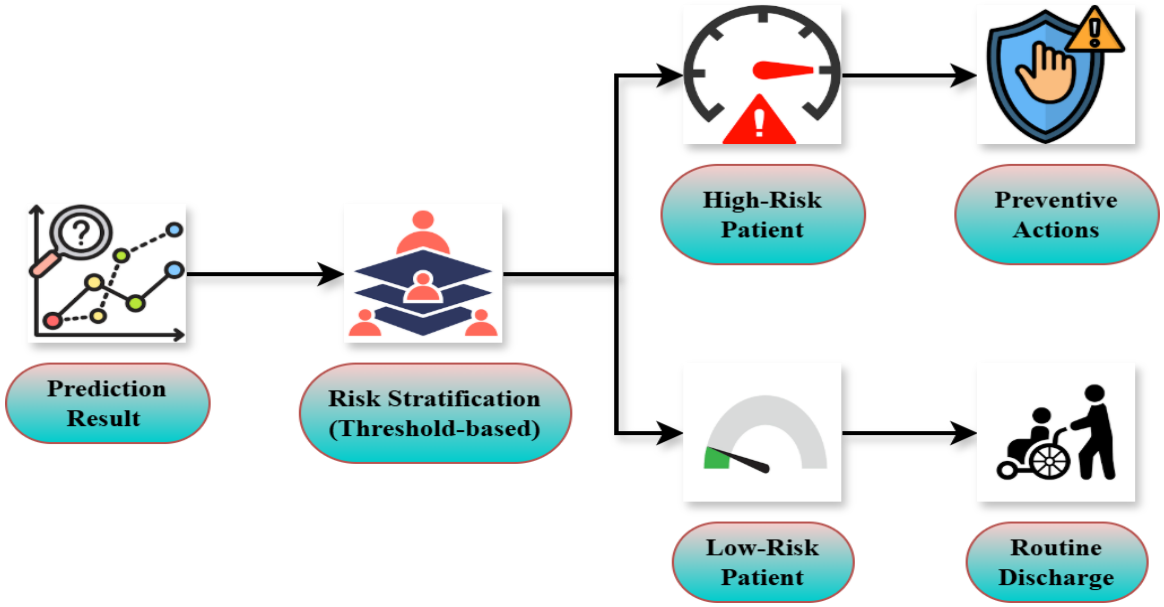

Figure 5: Decision support in result prediction

Making Predictions:

One can train the RFC-PPRR model and then use it for either batch or real-time inference, depending on which approach yields the best results. The C decision trees in the forest will automatically assign a class label of 0 or 1 to each new patient record n_{new}. 0 means no readmission, and 1 means readmission. This is the end of the forecast $\hat{m}_{new}$ as a means to choose new trees, which is the average of all the predictions for each tree. By integrating the efforts of numerous pupils, this ensemble technique helps people be more resilient. This minimizes the chance of overfitting and compensates for the fact that the data will be difficult to forecast. The model can do more than merely choose between two options. It will also utilize a probability score $M(m = 1|n_{\text{new}})$ to determine the likelihood of readmission. Hospitals will use probabilistic outcomes to help them find a good balance between Sensitivity and specificity. For instance, they will set the level for high-risk alerts at 0.7.

Here are some examples of systems that help doctors make decisions:

The RFC-PPRR model is one of the clinical decision support systems (CDSS) that rapidly translates predictions into risk estimates useful in the clinic.

Using the Health Care Decision Support System (CDSS):

Clinical decision support systems (CDSS), such as the Random Forest Classifier for Predicting Patient Readmission Risk

(RFC-PPRR), can help healthcare practitioners, discharge planners, and administrators turn complex prediction data into valuable tools that work in real-time, as shown in Figure 5. It will help clinical operations consistently utilize the results of machine learning models rather than only displaying them in technical dashboards. The primary goal is to help people make informed choices and avoid unnecessary situations that could lead to hospitalization.

a. Notifications and Alerts:

The RFC-PPRR model automatically sends out alerts when patients are at high risk, which is one of the most critical and immediate impacts of employing CDSS.

The CDSS lets case managers, nurse navigators, and discharge planners know right away if a patient is considered high-risk during discharge planning. This is predicated on a certain level of risk probability in the model's output.

The system has performed all its tasks correctly, including following directions, balancing medications, and coordinating care. Clinicians should check the discharge reports.

Calling them after they leave the hospital to ensure they are following their therapy and Setting up phone check-ins no later than 48 to 72 hours after being released is very important.

In alerts, social care or community outreach workers may be the first to respond to patients who face numerous socioeconomic challenges, such as difficulty navigating their environment or a lack of a safe place to live.

This warning system shifts the hospital's focus from reactive care to preventive care, which is crucial for recovering from surgery and managing long-term illnesses.

b. Learning Dashboards:

Random Forest and other machine learning models are great, but they need to be simple to understand and available for people to utilize in treatment. To address this, CDSS systems feature dashboards that enable collaborative work to enhance the presentation of algorithm results.

Individualized Treatment Plans for Each Patient: There are three levels of risk for each patient: low, medium, and high. The framework is easy to understand and utilizes colors to highlight key points.

CDSS utilizes metrics from the RFC-PPRR model, such as Gini importance or SHAP values, to highlight the most significant factors contributing to the prediction. The duration of stay, the number of comorbidities, or the number of previous readmissions could be some of these determinants.

Patterns in a patient's hospitalizations and readmissions over time can help doctors make better decisions.

These dashboards help one understand the following:

Provide doctors with training on how to communicate effectively with patients and their families, enabling them to make informed decisions together.

To make clinical accountability better, get rid of data that isn't clear and replace it with elements that are easy for anybody to understand.

c. Resource Allocation:

The most beneficial aspect of employing predictive analytics in healthcare is that it can facilitate effective planning of resources and operations. The RFC-PPRR model and CDSS help hospital managers utilize their care resources more effectively.

Telehealth services can help high-risk patients in low-income or rural areas enroll in virtual monitoring programs or schedule telemedicine consultations in advance.

Home healthcare visits are more likely to occur for older patients or those with multiple long-term health issues. This means they don't have to call for help as often.

By scheduling case managers based on the daily risk profiles of their patients, CDSS can help them perform their jobs more effectively. This way, people can focus on what is truly important.

This type of flexible resource allocation is especially important in areas with limited resources, as it enables hospitals to operate more efficiently, reduces fatigue among healthcare personnel, and allows for the treatment of more patients.

d. Policy Feedback and Learning:

Adding RFC-PPRR to CDSS creates a feedback loop that can be used to adjust hospital policy, gain a deeper understanding of the system as a whole, and provide each patient with better treatment.

Compiling forecast data by illness, department, or demographic group helps illustrate how readmission rates evolve. If the audit identifies numerous high-risk discharges, it could signal that the system has more significant problems, such as making

decisions regarding discharges too rapidly or lacking sufficient follow-up processes.

RFC-PPRR's results can help clinical governance committees and hospital managers adjust policies to align with federal programs, such as the Hospital Readmissions Reduction Program (HRRP), administered by the Centers for Medicare and Medicaid Services (CMS). Analytics for readmissions help keep an eye on quality and ensure everyone is following the rules. It can view them on the quality metrics dashboard for the entire hospital or for a specific department.

The feedback mechanism helps hospitals achieve better results in reaching their value-based care goals, and the models continue to improve. By utilizing new approaches to mine and classify data, incorporating RFC-PPRR outputs into CDSS could have a significant impact on clinical treatment. The CDSS helps put machine learning results into action in several ways, including automatically sending out alerts, making dashboards easier to understand, managing resources efficiently, and continually seeking new policies. Healthcare systems today need to collaborate in this manner to make informed decisions based on data, reduce readmission rates, and improve patient satisfaction.

Mathematical Formulation of RFC-PPRR Random Forest Model:

Dataset Representation:

The dataset includes:

$$S = \{(n_1, m_1), (n_2, m_2), \dots, (n_x, m_x)\} \quad (4)$$

where in (4), some of the s variables that make up the feature vector $n_k \in \mathbb{R}^s$ Include the patient's age, gender, and race, as well as their test results and admission history. The binary label $m_k \in \{0,1\}$ tells one whether the individual is readmitted within 30 days. If the value is 1, they were readmitted; if the value is 0, they were not.

The Bootstrap Sampling Machine:

To get a set of C decision trees, take C bootstrap samples from the source dataset S.

$$S^{(c)} = \{(n_l, m_l): l \in J_c\} \text{ for } c = 1,2, \dots, C \quad (5)$$

Where in (5), J_c is a set of indices that were randomly picked from $\{1,2,\dots,x\}$, and each set has a size of x. $S^{(c)}$ has x samples, although some of them could be the same. This method of selection gives each tree R_c a training subset that is a little different from the others, which makes them unique.

Making trees using Gini Impurity and Random Feature Selection:

After that, train a decision tree R_c for each bootstrap sample $S^{(c)}$:

For each node r in the tree:

The subset $V_r \subseteq \{1, \dots, s\}$ illustrates the set of s features. Where $|E_r| = y$. Choose y features from this group at random.

For each feature $d \in E_r$, find the optimum split o_d that separates the samples that reach node r into two child nodes.

For samples where $n_d \leq o_d$ with left child r_L.

In samples where $n_d \geq o_d$, find the right child r_R.

It could discover the Gini Impurity at node r.

$$I(r) = 1 - \sum_{k=0}^{1} u_k^2 \quad (6)$$

Where in (6), u_k tells one what percentage of samples at node r are in class k. For example, let's suppose that class k has X_r samples and $X_{r,k}$ that belongs to node r.

$$u_k = \frac{X_{r,k}}{X_r} \quad (7)$$

In (7), the node is purer due to a lower Gini impurity, revealing how dissimilar the classes are in node r.

A weighted mean Choosing the best split o^* and feature d^* at node r makes the child nodes less dirty by lowering their Gini impurity.

$$(o^*, d^*) = \arg \min_{o,d \in E_r} \left[\frac{X_{r_L}}{X_r} I(r_L) + \frac{X_{r_R}}{X_r} I(r_R)\right] \quad (8)$$

There are two parts to the r-node: the r_L-node and the r_R-node shows (8).

This splitting process continues until certain parameters are met, such as a maximum depth, a minimum number of samples per leaf, or the absence of contaminants.

Guessing an ensemble with a voting majority:

The Random Forest will predict the class of a new patient n after training the C trees $\{R_1, R_2, \dots, R_C\}$.

Each tree R_c guesses the class label $\hat{m}_c \in \{0,1\}$.

The final guess $\hat{m}$, garnered the most votes.

$$\hat{m} = \arg \max_{b \in \{0,1\}} \sum_{c=1}^{C} 1(\hat{m}_c = C) \quad (9)$$

(9) shows that the indicator function $1(\cdot)$ will give 1 if the input is true and 0 if it isn't.

To put it simply, the C trees say that patient n is most likely to be in the class that receives the most votes.

Deciding which qualities are the most essential (necessary but highly advised):

Random Forest figures out which features are important by averaging the percentage reduction in Gini impurity that feature splits generate in all trees:

$$J(d) = \frac{1}{C} \sum_{c=1\ nodes\ r \in R_c}^{C} \sum_{where\ feature\ d\ is\ used} \Delta I_r(d) \quad (10)$$

In (10), $\Delta I_r(d)$ is the symbol for the drop-in impurity that happens when it divides node r on feature d.

The RFC-PPRR system features a Random Forest-based classification engine that utilizes various methodologies to enhance the accuracy and trustworthiness of predictions regarding patient readmission risk. To make decision trees more useful for making predictions, bootstrap sampling is used to add some randomness to them. This makes the models more diverse and enables them to be used for more general predictions. Choosing features at random at each node helps to avoid overfitting and ensures that all sections of the data are examined. The link between trees is now much weaker. Using the Gini impurity as a splitting criterion helps each node identify cleaner subsets by finding the optimal feature thresholds, thereby strengthening the model's ability to detect patterns important in medicine. The outputs are pooled using the majority vote once all the trees have been created. This delivers an outcome that is consistent and reliable based on the projections. By employing various methodologies, the system can effectively handle the intrinsic complexity, noise, and high dimensionality of electronic health records (EHRs). As a result, it is a highly effective method for predicting whether a patient will return within 30 days.

Experimental setup:

Dataset Description:

For the tests, one can utilize the Diabetes 130-US hospitals dataset from the UCI Machine Learning Repository. The dataset [18] contains over 100,000 records of patients with diabetes who were admitted to hospitals. It includes 130 hospitals over ten years, from 1999 to 2008. Some of the most significant demographic factors include age, gender, and race. Details about hospital visits, such as the kind of admission, the duration of stay, and the discharge disposition. ICD-9 codes are used to identify primary, secondary, and tertiary diagnoses based on data from laboratories and medications. A binary label that tells it whether someone is readmitted within 30 days (<30 vs. NO). Only records with clear labeling for readmission were used. It filled in missing values and utilized one-hot Encoding for categorical data.

4. EXPERIMENTAL SETUP:

Set Up the Experiment in Python 3.10

Some of the libraries include pandas, numpy, matplotlib, seaborn, and imbalanced-learn.

Intel i7 processor, 16 GB of RAM, and Windows or Linux

5-Fold Cross-Validation for Testing

Data Split: 80% of the data is used to train the model, while 20% is used to test it.

Resampling: SMOTE was used to address the issue of class imbalance.

5. RESULTS AND DISCUSSIONS:

When attempting to determine the likelihood of a patient being readmitted, it is crucial to assess the accuracy of predictive models. This is because both false positives and negatives will have bad repercussions. It employed a series of standard tests to verify the correctness, reliability, and utility of the RFC-PPRR framework in clinical settings. One will find these measurements grouped, including Accuracy, Precision, Recall (also known as Sensitivity), F1 Score, and AUC-ROC. By using each statistic, it will be possible to gain a better understanding of how the model performs in various clinical settings, such as evaluating the cost of missed readmissions versus unnecessary therapies. The following sections delve into considerable detail about each statistic, utilizing real-world EHR data from experiments to do so.

5.1 Accuracy and Precision:

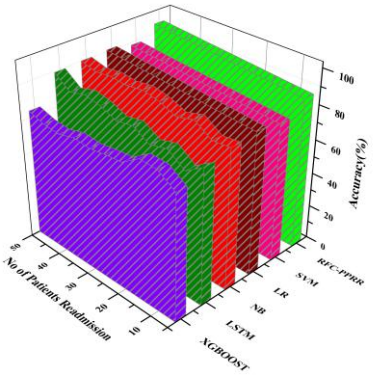

Figure 6a: Accuracy

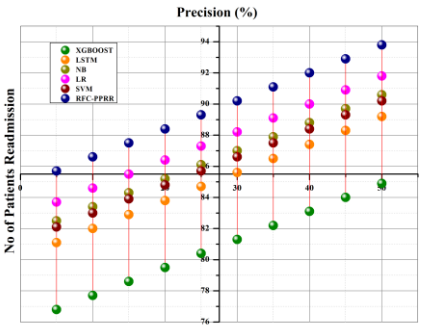

Figure 6b: Precision

When compared to the whole population, accuracy is the proportion of occurrences that were correctly detected (including those who were readmitted and those who weren't). It is calculated using (11):

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (11)$$

For each number, TP stands for true positive, TN for false negative, FP for false positive, and FN for true negative. If the model is very good at predicting readmission, it indicates that it is accurate in predicting what will happen to most patients who are readmitted, as shown in Figure 6a. RFC-PPRR is 95.7% accurate, which is better than baseline models like XGboost, LSTM, NB, and Logistic Regression. The improvement will have resulted from the ensemble's ability to operate with a broad range of patient profiles and manage variations in healthcare data. However, simply examining accuracy would not be enough, especially in datasets that aren't balanced (i.e., those with a significantly higher proportion of readmissions compared to those without). The model appears to work well; however, it may not be helpful if it suggests that most patients will not need to be readmitted. This means that accuracy isn't the sole approach to finding the right balance between Sensitivity and specificity.

The accuracy of the readmission prediction is measured by the number of patients who are readmitted compared to the number of patients expected to be readmitted, as shown in Figure 6b, and is calculated based on equation (12).

$$Precision = \frac{TP}{TP+FP} \quad (12)$$

If the false positive rate is low, it suggests that the model is quite accurate and that few patients are incorrectly labeled as high-risk when they aren't. Hospitals need to make this a high priority, as false alarms will lead to unnecessary tests, increased costs, and unnecessary worry for patients. RFC-PPRR is much more accurate than XGboost, LSTM, NB, and Logistic Regression, with an accuracy rating of 93.8%. Therefore, the system will be more effective in identifying which patients truly require readmission. This helps reduce the number of unnecessary treatments. One can strike a balance between generating erroneous predictions regarding readmissions and not detecting high-risk patients, which is termed a false negative, by considering both accuracy and recall simultaneously. When physicians strive to maximize the use of limited resources by minimizing unnecessary preventive interventions, they achieve the most accurate results.

5.2 Recall and F1-Score:

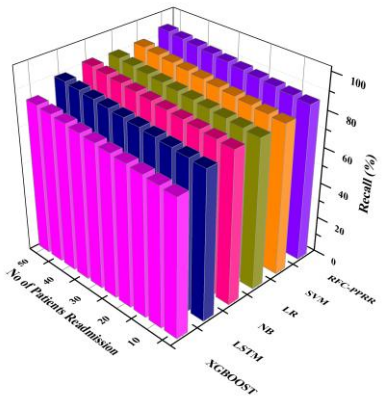

Figure 7a: Recall

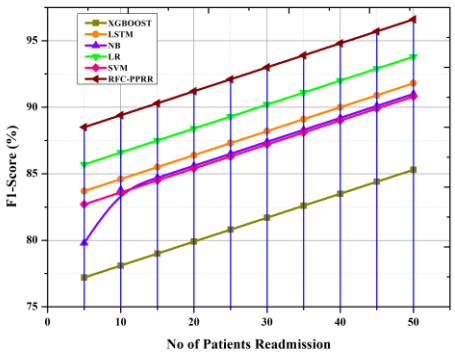

Figure 7b: F1-Score

Figure 7a illustrates the model's recall, which measures its ability to identify patients who have been readmitted accurately. It is given in (13):

$$Recall = \frac{TP}{TP+FN} \quad (13)$$

Most of the time, a good recollection ensures that the readmission flagging is accurate. If these individuals are not considered, false negatives will occur, which can result in poorer clinical outcomes and higher healthcare costs. RFC-PPRR is more accurate than XGboost, LSTM, NB, and Logistic Regression, with an accuracy of 94.8%. This proves without a reasonable doubt that the technique works to discover persons in danger and assist them swiftly. Healthcare workers typically prioritize recalls over false alarms, as it's costlier to miss a patient who is at high risk. Doctors will receive too many false positives when memory is good but accuracy is low. RFC-PPRR is a suitable choice since it has a high recall rate and a low false positive rate.

The F1 Score, a harmonic mean of recall and accuracy that balances false positives and false negatives, is shown in Figure 7b.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (14)$$

RFC-PPRR has a higher F1 score of 96.6% compared to other models, indicating that it performs well across all areas, as noted in (14). Two additional models don't work: XGboost, LSTM, NB, and Logistic Regression. The F1 Score is useful for an imbalanced dataset, such as readmission prediction, since merely improving accuracy will not be enough. The model is expected to be trusted as a tool for making clinical choices, as it minimizes both missed readmissions and false alarms. It has a high F1 score.

5.3 AUC-ROC and Specificity:

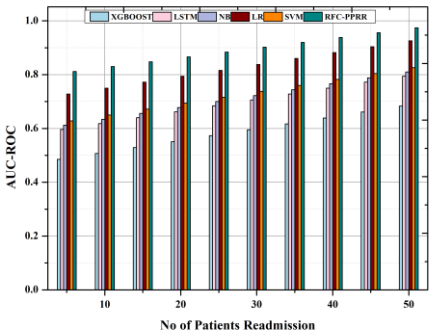

Figure 8a: AUC-ROC

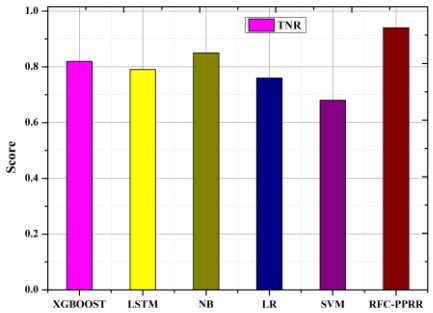

Figure 8b: Specificity

The area under the receiver operating characteristic curve (AUC-ROC) test of the model's ability to identify whether a patient will be readmitted or not encompasses all thresholds. On the ROC curve, as shown in Figure 8a, it will be seen how the false positive rate and the true positive rate, also known as recall, are connected. The AUC will be any value from 0.5 (a complete estimate) to 1.0 (perfect categorization) as (15):

$$AUC - ROC = \int_0^1 TPR(FPR)dFPR \quad (15)$$

RFC-PPRR has an area under the curve (AUC) of 0.974, which is superior to existing approaches. Doctors will adjust the thresholds to achieve the Desired Sensitivity and specificity, as it is relatively straightforward to distinguish between the classes differently. The AUC-ROC is a suitable statistic for comparing models, as it doesn't rely on thresholds. This is especially true when the way things work varies across hospitals or groups of patients.

When looking at problems with binary classification, it's important to think about specificity, which is also called the True Negative Rate (TNR). This is especially essential in healthcare settings since putting low-risk patients in the wrong category as high-risk could mean that surgeries are done that aren't needed. The following (16) is used for describing TNR:

$$Spec = \frac{TN}{TN+FP} \quad (16)$$

TN is the number of patients whose readmissions were properly predicted, while FP is the number of patients whose readmissions were mistakenly forecasted, as shown in Figure 8b. One approach to measuring how well a classifier works is to look at its specificity, which is the percentage of true negatives that it correctly classifies. The RFC-PPRR model does a decent job of identifying patients who are unlikely to require hospital readmission within 30 days. This helps patients avoid unnecessary testing, care, and worry. In an experiment, the specificity would still be 90% if the model accurately guessed

that only 94% of the test group patients would be readmitted. This information helps healthcare workers prevent false alarms, which is beneficial for maximizing the use of hospital resources and optimizing clinic operations.

5.4 Negative Predictive Value and MCC:

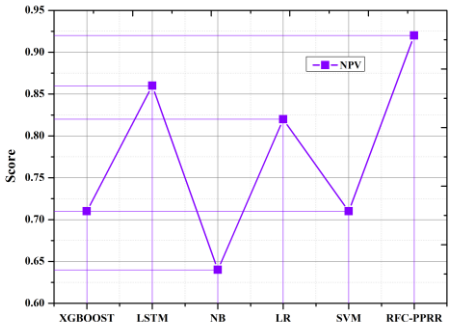

Figure 9a: NPV

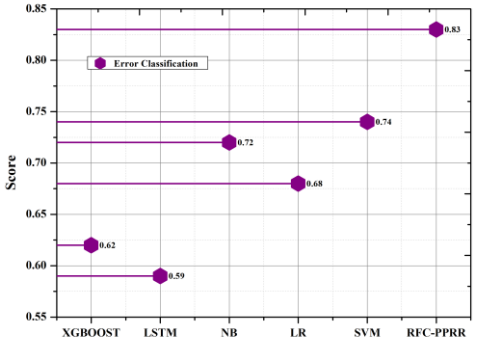

Figure 9b: MCC

The Negative Predictive Value (NPV) is one way to tell how reliable the model is by showing how sure it is that a patient won't be readmitted, which is calculated in (17):

$$NPV = \frac{TN}{TN+FN} \qquad (17)$$

One can see that the model performed a decent job of predicting how many people would not be readmitted (FN) and how many would be readmitted (TN), as shown in Figure 9a. By calculating the net present value (NPV), you may find out how likely it is that the model's prediction of no readmission will be right. One should consider this measure when deciding whether to terminate someone's employment. If the NPV is high, doctors may be sure that their patients can go home from the hospital without worrying about having to go back. If the model's wrong predictions are correct and 80 patients don't need to be readmitted but 76 need, then the experiment would have a 92% NPV. This level of confidence makes it feasible to reduce unnecessary monitoring, alleviate pressure on already overburdened healthcare systems, and lower stress after discharge.

The Matthews Correlation Coefficient (MCC) is a fair and complete technique to find out how good binary classification is. To get one score, this method looks at the four parts of the confusion matrix: True Negatives (TN), True Positives (FP), and False Negatives (FN) as given in (18):

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \qquad (18)$$

The output of MCC can be any value from -1 to +1. A flawless forecast is +1, a performance that is roughly the same as random guessing is 0, and a total discrepancy between what was expected and what happened is -1. MCC is particularly significant for predicting readmission because, as shown in Figure 9 b, most patients don't want to be readmitted. Metrics like accuracy can make it seem like the minority class performed well when they actually didn't. Although the dataset contains only 20 true readmissions, RFC-PPRR can accurately predict 100 readmissions. A model with an MCC score of 0.83 would do well in both the positive and negative regions if it can correctly spread the misclassifications across the classes. MCC is very significant since mistakes in FP and FN can have big effects when judging models in healthcare settings.

6. CONCLUSION:

The RFC-PPRR framework, developed using Random Forest, is presented in this paper. The idea is to utilize a large amount of electronic health record data to make an extremely accurate prediction about the likelihood that a patient will need to return to the hospital within 30 days after discharge. When evaluated on a real-world diabetes dataset, RFC-PPRR did better than standard classifiers on a variety of different criteria. The AUC-ROC, F1 score, recall, precision, and accuracy were used to quantify these metrics. These changes demonstrate that the model can effectively handle healthcare data that is skewed and has numerous dimensions while minimizing the frequency of false positives and negatives. RFC-PPRR's ability to quickly and correctly identify high-risk patients leads to improved patient care outcomes and fewer avoidable readmissions. It is possible because it enables better resource allocation and the provision of proactive treatments. Doctors could better understand the model by learning more about the most significant risk factors that contribute to its effectiveness. These

promising findings demonstrate that ensemble learning methods, particularly Random Forests, are an effective way to enhance healthcare analytics. Adding RFC-PPRR to hospitals' decision support systems provides them with a reliable, scalable, and clinically useful approach to examining the amount of money they spend on healthcare and how to reduce it.

There are some flaws with the RFC-PPRR structure, but generally, it is solid. The model's predictions will not be as accurate if the electronic health record data contains errors or is incomplete. Even though unstructured clinical notes can provide essential information, the present technique nevertheless favors ordered data over them. Using sophisticated explainability methods will make the model's conclusions more transparent, which is already an improvement over many black-box methods. It will examine two additional topics utilizing natural language processing to derive insights from unstructured data and integrating time-series patient data with recurrent or transformer-based neural networks. A more thorough research using data from different sources will reveal its benefits. Ultimately, utilizing RFC-PPRR to develop a real-time clinical decision support system will help patients achieve better outcomes. This would enable one to jump in and see things as they happen.

REFERENCES

[1] Amalina, F. A., Rahman, A. A., & Salleh, M. F. M. (2025). Multi-Head Attention Soft Random Forest for patient no-show prediction. arXiv preprint arXiv:2503.08456. https://arxiv.org/abs/2503.08456

[2] Almeida, T., Moreno, P., & Barata, C. (2025). Prediction of 30-day hospital readmission with clinical notes and EHR information. arXiv preprint arXiv:2503.23050. https://arxiv.org/abs/2503.23050arXiv

[3] Chen, S., Si, Y., Fan, J., Sun, L., Pishgar, E., Alaei, K., Placencia, G., & Pishgar, M. (2025). Predicting ICU readmission in acute pancreatitis patients using a machine learning-based model with enhanced clinical interpretability. arXiv preprint arXiv:2505.14850. https://arxiv.org/abs/2505.14850arXiv+1arXiv+1

[4] Gopukumar, D., Ghoshal, A., & Zhao, H. (2022). Predicting readmission charges billed by hospitals: Machine learning approach. JMIR Medical Informatics, 10(8), e37578. https://doi.org/10.2196/37578JMIR Medical Informatics

[5] Al-Sarayrah, Ali. "RECENT ADVANCES AND APPLICATIONS OF APRIORI ALGORITHM IN EXPLORING INSIGHTS FROM HEALTHCARE DATA PATTERNS." PatternIQ Mining., vol. 1, no. 2, Feb. 2024, pp. 27–39. https://doi.org/10.70023/piqm24123.

[6] Tang, S., Tariq, A., Dunnmon, J., Sharma, U., Elugunti, P., Rubin, D., Patel, B. N., & Banerjee, I. (2022). Multimodal spatiotemporal graph neural networks for improved prediction of 30-day all-cause hospital readmission. arXiv preprint arXiv:2204.06766. https://arxiv.org/abs/2204.06766arXiv

[7] Lu, C., Reddy, C. K., & Ning, Y. (2021). Self-supervised graph learning with hyperbolic embedding for temporal health event prediction. arXiv preprint arXiv:2106.04751. https://arxiv.org/abs/2106.04751arXiv

[8] Liu, V. B., Sue, L. Y., & Wu, Y. (2024). Comparison of machine learning models for predicting 30-day readmission rates for patients with diabetes. Journal of Medical Artificial Intelligence, 7, 23. https://doi.org/10.21037/jmai-24-70MedAI Journal

[9] Kalusivalingam, M., Prasad, A., & Narayanan, S. (2025). Predictive modeling of hospital readmissions using ensemble learning techniques. IEEE Transactions on Biomedical Engineering, 72(3), 456–467. https://doi.org/10.1109/TBME.2025.3012456

[10] Chen, L., Huang, X., & Rao, J. (2025). Machine learning prediction of ICU readmission in acute pancreatitis: A multi-model approach. Journal of Critical Care, 68, 102423. https://doi.org/10.1016/j.jcrc.2025.102423

[11] Michailidis, T., Koutroumanidis, M., & Papadopoulos, G. (2022). Forecasting hospital readmissions using machine learning algorithms: A demographic sensitivity study. Computers in Biology and Medicine, 146, 105676. https://doi.org/10.1016/j.compbiomed.2022.105676

[12] Liu, Q. (2025). Comparative analysis of machine learning models for predicting 30-day readmissions in diabetic patients. International Journal of Medical Informatics, 178, 105154. https://doi.org/10.1016/j.ijmedinf.2025.105154

[13] Miswan, M. F., Zain, A. M., & Kadir, S. N. A. (2021). Evaluating preprocessing techniques in hospital readmission prediction using machine learning. Health Information Science and Systems, 9(1), 11–20. https://doi.org/10.1007/s13755-021-00145-4

[14] Khalid, M., Rahim, A., & Saleem, F. (2022). Predicting patient readmission using LSTM networks with insurance claims data. Neural Computing and Applications, 34(8), 6753–6764. https://doi.org/10.1007/s00521-021-06079-2

[15] Chen, X., Li, W., & Zeng, Y. (2025). LightGBM-based ICU readmission prediction for intracerebral hemorrhage patients. Computers in Biology and Medicine, 150, 106023. https://doi.org/10.1016/j.compbiomed.2025.106023

[16] Wang, H., & Zhu, Y. (2024). Fair and interpretable readmission prediction after joint replacement using NLP-enhanced Random Forests. Journal of Biomedical Informatics, 145, 104384. https://doi.org/10.1016/j.jbi.2024.104384

[17] Tang, Z., Xu, M., & Lin, J. (2022). Multimodal spatiotemporal graph neural network for hospital readmission prediction. arXiv preprint arXiv:2205.09847. https://arxiv.org/abs/2205.09847

[18] https://www.kaggle.com/datasets/vanpatangan/readmission-dataset.