Machine Learning Based Predictive Framework for Intensive Care Unit Readmission

By:

Nadia Rasheed

(Registration No.: MS-CSE-21-361455)

Supervisor

Dr. Muhammad Usman Akram

DEPARTMENT OF COMPUTER & SOFTWARE ENGINEERING,
COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING,
NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY,
ISLAMABAD
December 27, 2023

Annex A

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS/MPhil thesis written by NS **Nadia Rasheed** Registration No. 00000361455, of College of E&ME has been vetted by undersigned, found complete in all respects as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial fulfillment for award of MS/MPhil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the thesis.

Signature : Usman

Name of Supervisor: **Dr Muhammad Usman Akram**

Date: 27-12-23

Signature of HOD:
(Dr Usman Qamar)
Date: 27-12-23

Signature of Dean: Rashid
(Brig Dr Nasir Rashid)
Date: 27 DEC 2023

Machine Learning Based Predictive Framework for Intensive Care Unit Readmission

By:

Nadia Rasheed

(Registration No.: MS-CSE-21-361455)

A thesis submitted to National University of Sciences and Technology, Islamabad
in partial fulfillment of the requirements for the degree of
Master of Sciences in Software Engineering

Supervisor

Dr. Muhammad Usman Akram

DEPARTMENT OF COMPUTER & SOFTWARE ENGINEERING,
COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING,
NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY,
ISLAMABAD

December 27, 2023

Dedicated to my exceptional parents and adored brothers whose tremendous support and cooperation led me to this accomplishment!

ACKNOWLEDGEMENTS

All praise and glory to Almighty Allah (the most glorified, the highest) who gave me the courage, patience, knowledge and ability to carry out this work and to persevere and complete it satisfactorily. Undoubtedly, HE eased my way and without HIS blessings I can achieve nothing.

I would like to express my sincere gratitude to my advisor Dr. Muhammad Usman Akram for boosting my morale and for his continual assistance, motivation, dedication and invaluable guidance in my quest for knowledge. I am blessed to have such a co-operative advisor and kind mentor for my research.

Along with my advisor, I would like to acknowledge my entire thesis committee: Col. Aqib Perwaiz, Assistant Professor Jahan Zeb and Lecturer Anum Abdul Salam for their cooperation and prudent suggestions.

My acknowledgement would be incomplete without thanking the biggest source of my strength, my family. I am profusely thankful to my beloved parents who raised me when I was not capable of walking and continued to support me throughout in every department of my life and my loving brothers who were with me through my thick and thin.

Finally, I would like to express my gratitude to all my friends and the individuals who have encouraged and supported me through this entire period.

ABSTRACT

The cost of intensive care is huge, which necessitates careful thought regarding transfer of patients to lower-level ward care. Discharging a patient too early carries the risk of inadequate monitoring and care, often leading to readmission to the ICU. This risk can be mitigated by state-of-the-art machine learning methods. Limited research was carried out on readmission prediction tasks and the methods used were unable to attain good results. This study focuses on developing an ICURP (Intensive Care Unit Readmission Prediction) framework that can be used for the effective prediction of unplanned ICU readmission within 30 days. Particularly, the framework deals with the missing values (via the last observation carried forward technique) and data imbalance (via the Over-sampling Technique) problems. Our approach incorporates temporal features from chart events data with low-dimensional embeddings of medical concepts such as diseases coded using the ICD-9 code. Convolutional neural network (CNN) is used to fit three alternative CNN models using last 24-hour, 48-hour and 72-hour ICU stay data. Models are trained and validated using the Medical Information Mart for Intensive Care (MIMIC-III) dataset. To evaluate the effectiveness of our proposed methods, we conducted testing on the unseen data of the MIMIC-III dataset. The model trained using the last 48-hour ICU data has outperformed other models and reached an area under the curve of receiver operating characteristic (AUC-ROC) of 0.88. To establish a comparison, two Recurrent Neural Network (RNN) based models Long-short-term-memory (LSTM) Gated Recurrent Unit (GRU) and four conventional models (SVM, LR, NB, KNN) are trained using ICU data. The results suggested that our ICURP framework has the potential to surpass the existing standard of ICU discharge by accurately predicting readmissions up to 30 days of discharge time using a reduced features set.

Key Words: readmission prediction, intensive care unit (ICU), convolutional neural network (CNN), machine learning (ML), time series analysis.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

Introduction

Intensive care units (ICUs) deliver the most specialized care within hospital settings, catering to patients with severe medical conditions. Due to the limited ICU resources, only those patients are kept under ICU treatment that needed it badly. Afterwards, patients are discharged from ICU and transferred to low level wards [1]. Unplanned ICU readmission refers to situations where a patient is discharged from the ICU to a non-critical care hospital ward but later returns to the ICU on an unplanned basis. This situation is observed in approximately 4% to 11% of ICU patients and is linked to a significant 6 to 7 fold increase in the risk of mortality [2]. The readmission of patients to the ICU poses other financial risks as well as risks to morbidity and mortality [3, 4]. Discharging patients from ICU too soon may potentially subject them to the risks of inadequate treatment, ultimately leading to mortality [5]. According to reports, the mortality rates for patients who are readmitted vary, spanning from around 26% to 58%. Shockingly, developed countries hospitals also face high rates of readmission to the ICU, with approximately 10% of patients being readmitted during their hospital stay. Furthermore, there is an alarming upward trend in the United States, where ICU readmission rates have increased from 4.6% in 1989 to 6.4% in 2003 [6, 7].

To resolve this problem, the Affordable Care Act (ACA) developed the Hospital Readmissions Reduction Program in 2010. This program imposed the penalties on hospitals that have higher than expected readmission rates within 30 days [8]. Since the implementation of the Hospital Readmissions Reduction Program on October 1, 2012, hospitals have faced significant financial penalties which amounted nearly $2.5 billion for readmissions, with an assessed $564 million in financial year 2018 alone [9]. According to an estimate, 18% to 22% of unplanned readmissions are avoidable [10]. In order to minimize preventable readmissions, hospitals must identify patients who are at an increased risk of ICU readmission [11]. Decisions regarding ICU discharge heavily rely on subjective clinical judgment, making them susceptible to human errors. Factors like hospital crowding, staff availability, and caregiver fatigue, seams unrelated to the patient's well-being, can also impact the timing of discharge. Consequently, the lack of standardized and patient-centered criteria in ICU discharge decisions raises concerns about the accuracy of these decisions.

Utilizing a machine learning model can enhance the decision-making process during discharge. This approach alleviates the cognitive burden placed on caregivers by providing them with objective and standardized risk assessments. ML models can be constructed using numerous datasets, including electronic health records (EHRs), administrative claims, and insurance claims. An overview of 26 distinct prediction models for readmission is presented in a systematic review of readmission prediction in [12]. Out of these 26 models, 23 models utilized electronic health records (EHRs) as the primary data.

Despite the existence of several studies aimed at finding high risk readmission patients, we still face challenges in achieving a practical solution. These studies reveal multiple limitations. Few predictive models have a limited scope, as they solely focus on specific diseases rather than providing a generalized solution. For example, certain studies targeted heart failure [13-16], diabetes [17], HIV [18], or kidney transplants [19]. None of the existing models have been able to achieve a satisfactory level performance in predicting ICU readmissions [20]. Majority of studies haven't used sequential data, and information loss occurs if time series nature of data is not used [21].

This study focuses on developing an ICURP (Intensive Care Unit Readmission Prediction) framework that can be used for unplanned ICU readmission prediction, leveraging advanced deep learning techniques with the time series nature of the data. Our approach incorporates temporal features from chart events data with low-dimensional embeddings of medical concepts such as diseases coded using the ICD-9 code. Convolutional neural network (CNN) is used to fit three alternative CNN models using last 24-hour, 48-hour and 72-hour ICU stay data. The embeddings are used to reduce the sparse nature of dataset [22]. The proposed ICURP framework is trained and validated using the MIMIC-III dataset [23], which comprises a vast amount of information over 40,000 patients, including 60,000 ICU admission records and spanning over 10-year period. For evaluating the effectiveness of ICURP, we conducted testing on the unseen data of same MIMIC-III dataset. The model trained using last 48-hour ICU data has outperformed other models and reached an area under the curve of receiver operating characteristic (AUC-ROC) of 0.88. Our hypothesis suggests that our ICURP framework has the potential to surpass the existing standard of ICU discharge by accurately predicting readmissions up to 30 days from the discharge time. Block diagram of ICURP framework is shown in Figure 1.1.

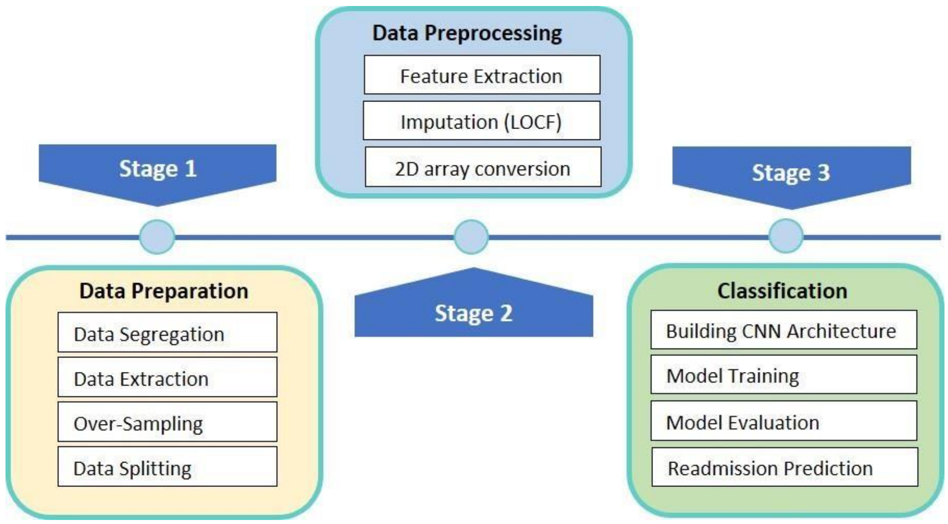

Figure 1. 1: ICURP (Intensive Care Unit Readmission Prediction) Framework

Motivation

Modern ICUs are best at providing constant monitoring of critically ill patients. These patients are liable to many complications which in turn will impact morbidity and mortality. The environment in ICU demands a significant staff-to-patient ratio and generates a substantial amount of data. The interpretation of real-time data and decision-making poses a terrible challenge for clinicians. However, the application of ML techniques in ICUs has shown promising advancements in detecting high-risk events at an early stage. By leveraging ML techniques, clinicians can better analyze and interpret ICU data, enabling more timely and informed decision-making. These advancements are made possible by the increasing computational power and the availability of openly accessible datasets like MIMIC-III [24].

The cost of intensive care is huge, which necessitates careful thought of when patients should be transferred to lower-level ward care. This decision is crucial to optimize the allocation of resources. However, discharging a patient from the ICU too early carries the potential risk of inadequate monitoring and care, often leading to readmission to the ICU. Therefore, it is important to strike a balance in determining the appropriate timing for step-down, ensuring that patients receive optimal care while minimizing the likelihood of readmission.

Problem Statement

Accurate and timely prediction of readmission has much significance for ICU patients. During literature survey, we came to the conclusion that limited research carried on readmission prediction task and the methods used were unable to attain good results. The review also emphasized the usage of clinical notes and time series data as these are critical in decision making. The purpose of this research is to explore latest learning techniques to improve prediction performance. This will focus on identifying critical factors associated with patients ICU readmission and develop a predictive framework using advanced data mining and machine learning techniques.

Aims and Objectives

Major objectives of this research are as follow:

- Review and comparison of recent developments for patients in ICU. Reviewing the latest techniques to determine when a patient should be discharged or transferred to low level ward.
- To study all the factors involved in ICU readmission.
- To perform comparison between the conventional machine learning techniques and the latest deep learning techniques available.
- To develop a 30 days ICU readmission predictive framework which outperform other state-of-the-art techniques by leveraging the ML techniques so that clinicians can better analyze and interpret ICU data, enabling more timely and informed decision-making.
- Particularly handling data imbalance and missing values through over-sampling and last observation carried forward technique respectively. Moreover, using optimum

features to reduce computational complexity.

- The research also focuses on the sequential and non-sequential way of handling features and analyzes the impact of combining different features like chart events data with ICD-9 diagnosis embeddings. By using the advanced ML architectures trained on MIMIC-III dataset, the implementation of our approach aspires to optimize resource allocation, improve patient outcomes, and enhance the overall efficiency of ICU care.

Structure of Thesis

This work is structured as follows:

Chapter 2 covers the background detail of topic and gives review of the literature with respect to the researchers' momentous work done in recent past years for the ICU readmission prediction task.

Chapter 3 covers the proposed methodology in detail. It introduces database first and includes step wise explanation of all techniques involved for the development of prediction framework.

Chapter 4 covers all the experiments starting from conventional techniques to the latest techniques. Results of all experiments are discussed in detail with the figures plots and tables.

Chapter 5 gives the comparative analysis for results of all experiments performed in previous section. Moreover, it also compares the results with the state-of-the-art.

Chapter 6 concludes the thesis and discloses future scope of this research.

Chapter 2

PRELIMINARIES & LITERATURE REVIEW

Background

Intensive care units: An intensive care unit is a department in any hospital that is specially allocated to provide intensive medication and care. The name ICU also termed as critical care unit (CCU) or intensive treatment unit (ITU). To ensure patients' normal body functioning, patients at higher risk of life threat or having severe injuries are kept under constant monitoring. ICUs are also different from general hospital wards as they carry advanced medical resources and equipment which are not routinely available in other hospital wards. Moreover, they also require a higher trained staff including physicians, therapist and nurses who got training in caring critically ill patients [1]. Patients with unstable conditions can be referred directly to ICU from ward or from emergency department. Those patients are also referred who have gone through critical surgery and need intensive care afterwards. Hospitals are equipped with a range of specialized ICUs adapted to meet special patient conditions or medical needs. Few of them are: Coronary Care Unit (CCU), Medical Intensive Care Unit (MICU), Cardiac Surgery Recovery Unit (CSRU), Surgical Intensive Care Unit (SICU) and Trauma Surgical Intensive Care Unit (TSICU). Depending on the hospital's demography and financial conditions, a hospital can allocate percentage of total beds for intensive care. As intensive care is expensive, United States hospitals allocate up to 20% of total beds as intensive-care beds. Whereas, hospitals in United Kingdom allocate only up to 2% of total beds as intensive-care beds [25].

Machine learning techniques: Machine learning is a sub field of artificial intelligence which introduced the concept of learning from data, enabling systems to make informed decisions according to their learning. Before this concept, systems were performing according to what they were programmed for. Machine learning has three main types: supervised learning, unsupervised learning and reinforcement learning [26]. These are explained below along with other types as well:

- Supervised Learning: The ML model is trained on labeled data, where each data sample is coupled with a target or outcome. The model learns from the input features and consequent target values to make predictions on unseen data. Classification and regression problems are solved through supervised leaning techniques.
- Unsupervised Learning: This technique involves training the model on data which is not labeled and the algorithm identifies patterns and structures within the data without any defined target label. Clustering, anomaly detection, and dimensionality reduction problems are solved through unsupervised learning techniques.
- Semi-supervised Learning: It combines both labeled and unlabeled data for training. It leverages the limited labeled data to guide the model's learning from the larger pool of unlabeled data.
- Reinforcement Learning: This technique employs training an agent to interact with an environment and learn optimal actions. The agent receives rewards or feedback based on its actions.
- Deep Learning: It is a subfield of machine learning that uses artificial neural networks, specifically deep neural networks with multiple layers. These networks can automatically learn hierarchical representations of data and deep features. The automation of feature learning has made deep leaning achieve highest performance on solving complex problems. However, deep learning demands systems with high computational power. Convolutional neural networks (CNNs) and recurrent neural networks (RNNs) are specialized types of deep neural networks (DNNs).

Each machine learning technique has its pros and cons, making them appropriate for different types of problems. The choice of technique varies depending on the nature of the data, the available resources, and the specific goals of the task.

Time series data: Time series data refers to a series of data points collected over consecutive time intervals. It is the representation of how a variable or a set of variables change over time. In time series data, the order and time interval between data points are important, as they capture the temporal dependencies and patterns present in the data. Time series data can be observed in various domains like finance, economics, weather forecasting and stock market analysis [27]. Time series data has the following components:

- Temporal Order: The data points are arranged in sequential order, with each observation occurring at a specific time or time interval.
- Time Dependence: Dependency lies between adjacent data points. The value at a given time point can be influenced by past values.
- Seasonality: Exhibits recurring patterns over fixed intervals of time.
- Trend: Data may have a long-term upward or downward movement known as a trend.
- Irregularity and Noise: Due to measurement errors or external factors, data can be subject to random fluctuations, irregularities, and noise.

Time series data can be analyzed though following techniques:

- Descriptive Analysis: Examination through visual plots, and charts to identify patterns and trends.
- Segmentation: Separating the time series into its components, such as trend and, seasonality.
- Statistical Models: Application of statistical models like autoregressive integrated moving average (ARIMA), or state space models, and exponential smoothing models to predict the behavior.
- Classification: Using machine learning models, such as CNNs, RNNs, LSTM, or GBMs to learn complex patterns and make predictions while training models on historical data.
- Forecasting: Forecast future values, trends and behavior of the time series.

These techniques help to analyze time series data to provide valuable insights and support decision-making.

Readmission prediction: ICU hosts patients who need continues monitoring and care because of their critical health condition. Since resources in ICU are limited and costly, patients must be transferred to low level wards when their condition is relatively stable and are able to get regular treatment in other hospital wards. This discharge or down-transfer-time from ICU is very critical as if a patient is down-transferred early then patient's condition can deteriorate and he can return back to ICU. This scenario can cause a patient to stay longer in ICU and utilize more resources contrary to previous stay. While, if a patient would be discharged late, someone else can be deprived of ICU treatment. So, researchers are finding and optimizing automated solutions to predict when a patient should be discharged from ICU. Some researchers have focused patients' readmission within 30 days of their discharge, others on within 2 days of discharge etc. Our research is also targeted on the same problem of predicting patient ICU readmission within 30 days. It means if a visit is followed by another visit within 30 days or less, then the untimely discharge of the patient from the hospital based on earlier visit instigated the readmission. In addition, if a patient dies within 30 days from discharge date, then that visit is also labeled as readmission. Based on patient's flow in and out of ICU, readmission cases are depicted in [28] and we have shown the same in Figure 2.1.

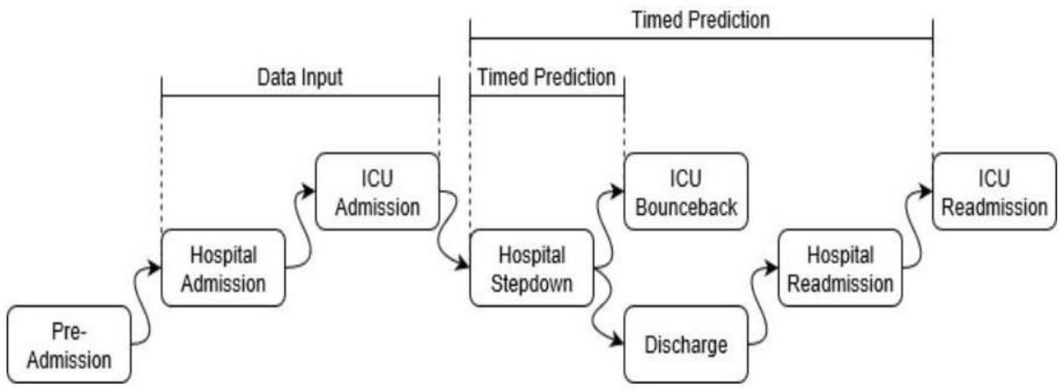

Figure 2. 1:Readmission Cases Scenarios

Literature Review

The main objective of this chapter is to extract and structure research findings in healthcare employing machine learning approaches, followed by the identification of potential avenues for future research. As this carry great importance and have prospective impact, we aimed to address the following review questions:

- Understanding the readmission phenomenon and its negative impact on hospital facilities.
- How are various healthcare data types represented as input for ML methods?
- How do time series data impact the predictive performance of model?
- Which ML models are most effective?
- Is there any impact of diagnosis specific readmission cases on model performance?

To answer these questions, we identified core characteristics including database used, input features, modeling methodology, nature of data (time series or other), and benchmark technique from each study. Table 2.1 elaborates these core characteristics one by one. The research papers have been critically reviewed on the basis of these mentioned characteristics. The survey includes papers based on both public and private datasets. Detail analysis of all studies will be given in 'Discussion' section of this chapter. This survey focuses on the papers published from major publishers like Elsevier, Springer, IEEE, Science Direct etc. from 2018 to 2023.

Table 2. 1: Core Characteristics and their Description

Feature	Description
Approach	Conventional ML model or Deep Learning model
Modeling	How features are modeled i.e., sequential or non-sequential modeling
Year	Determines the year in which study got published
Database	Determines the source of data like EHR, MIMIC etc.
Diagnosis specific readmission	Determines whether the focus is on studying patients having certain disease like heart failure patients etc.
Nature of data	Time series or other (static data)
Readmission after days discharge	Determines the days considered after patient's discharge
ML methodology	Models or architectures used for prediction task
Benchmark	State of the art techniques which outperformed

Classification of Research Articles

To present the summaries of all studies here, the articles included were reviewed and classified into two categories as shown in Figure 2.2. Conventional Models incorporates traditional modeling way which is typically composed of two steps, 1) feature engineering and 2) model building. Feature engineering extracts features from data that are good for the model building. Contrary to this method, Deep learning models supports an end-to-end learning mechanism by integrating the feature engineering process implicitly in the learning pipeline [29].

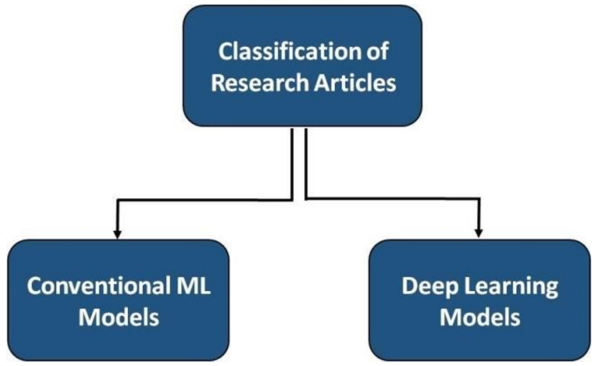

Figure 2. 2: Classification of Research Articles

- **Conventional ML Models**

In 2018, Rojas, J.C., et al. [30] developed an ICU readmission prediction model using ML techniques and compared it to previously published algorithms. From, University of Chicago, 24,885 ICU transfers to the wards were included; with 60% in the training and 40% in the internal validation. For external validation; 42,303 ICU admissions from MIMIC-III were considered. Patients' considered features were: nursing assessments, Ninth Revision codes from prior admissions, ICD, medications, ICU interventions, diagnostic tests, vital signs, and laboratory results. These were used as predictor variables and fed in gradient-boosted model. The Stability and Workload Index for Transfer score and the Modified EarlyWarning Score were also used to evaluate the accuracy. AUROC score of 0.76 achieved from ML derived model which is significantly better than the Stability and Workload Index for Transfer score and Modified Early Warning Score.

In 2018, Pakbin, A., et al. [28] have worked on identification of readmission risks for patients who got discharged from ICU within 24-hours, 72-hours, 7-days, 30-days, and also for the bounceback readmissions(readmission in the same hospital admission). MIMIC-III database is being used here and 58 features from the Chartevents, 2,784 features from Labevents and demographic data were extracted. Mean, standard deviation, minimum, maximum, last measurements, and the count was used to represent time series data. Due to the limit set on number of days, different sample size sets got generated like for 24 hrs 998 samples, 72 hrs 2206 samples, 7 days 3559 samples, 30 days 6235 samples and for bounceback they got 3637 samples. Data was split with 80, 20 ratios for training and testing two models Logistic Regression and XGBoost and the results validated using stratified 5-fold cross validation. XGBoost has performed better than Logistic Regression with AUROC of 0.76.

In 2019, McWilliams, C.J., et al. [31] used two datasets which were created using historical data; 7592 samples from the MIMIC-III and 1870 samples from the Bristol General Intensive Care were taken. A RF and a LR that had been trained using multiple source cross-validation, both outperformed the initial criteria and generalized well across cohorts. The classifiers had good agreement on the characteristics that were most indicative of readiness for discharge.

In 2019, Min, X., B. Yu, and F. Wang [29] realized the need to predict the 30 days readmission risk for Chronic Obstructive Pulmonary Disease (COPD) patients because of prevailing chronic pulmonary condition. Medical claims dataset containing 111,992 patients' records from the Geisinger Health System was used to evaluate both types of models. These patients were monitored from January 2004 to September 2015. Knowledge driven and data driven features are extracted from dataset during feature engineering process and these are fed in five models for performance evaluation. Models include Logistic regression and its variants, Random Forest, Support Vector Machine, Gradient Boosting Decision Tree and Multi-Layer Perceptron. While CNN, RNN, LSTM and GRU are evaluated on deep features. By combining both features, GBDT outperformed with AUROC score of 0.653 on one year claims history dataset.

In 2019, Li, Z., et al. [32] have determined the likelihood of ICU readmission by creating data-driven predictive algorithms. NLP techniques were used to carefully depict the discharge report for each hospital admission from MIMIC-III dataset. Bag-of-words (BoW) was used as the vectorization technique and Unified Medical Language System (UMLS) was used to standardize inconsistent nature of discharge summaries by uniquely identifying medical concepts (CUIs). Hence, two features set 1) Bag-of-Words and 2) Bag-of-CUIs were generated. Five ML classifiers (logistic regression, support vector machine, random forest, gradient boost Decision trees, and Naive Bayes) were trained on the two feature sets separately to perform prediction task. Logistic regression trained on Bag-of-CUIs had the best AUC score of 0.748 among other classifiers.

In 2020, Assaf, R. and R. Jayousi [33] predicted 30-day hospital readmission using multiple conventional ML algorithms like Multi-Layer Perceptron, Support Vector

Machine, Logistic Regression and, Random Forest. MIMIC-III dataset was used to train and test all above mentioned algorithms. Authors didn't consider ICU transfers but instead targeted only the discharge data from hospital which resulted in 3,323 positive samples and total dataset comprised of 6642 samples. Time series measurements for chart events vital signs data (Respiratory rate, Glucose level, Heart rate, Systolic blood pressure, Diastolic blood pressure, Body temperature, Weight), ICD codes for diagnosis and gender from the demographics data are the main extracted features. For each visit, min, max and mean values computed for chart event features against all time available measurements since duration of these measurements is not mentioned; while embeddings for ICD-9 code were aggregated through mean value calculation. All features are concatenated and used to train and test all models. Author suggested Random Forest as the best model for prediction as it has performed well with 0.65 accuracy and 0.66 area under the curve score.

In 2020, Liu, W., et al. [34] performed the 30-day readmission prediction task for patients hospitalised with acute myocardial infarction (AMI), congestive heart failure (HF), and pneumonia (PNA). In their study, they compared four prediction models for unplanned patient readmission using data from the Nationwide Readmissions Database (NRD). Three datasets related to each diagnosis are constructed to evaluate performance of four models i.e., Logistic Regression, eXtreme Gradient Boosting, ANN model using a feed-forward neural network and ANN model with deep structure. They demonstrate that the prediction task can be improved by combining ICD diagnosis and procedures codes embeddings from unsupervised Global Vector for Word Representations with artificial neural network classification models. In comparison to hierarchical logistic regression, proposed model raised the AUC for prediction of 30-day readmissions for AMI from 0.68 to 0.72, HF from 0.60 to 0.64, and PNA from 0.63 to 0.68.

In 2021, Moerschbacher, A. and Z. [35] have performed 30-day readmission prediction task on three different data types. MIMIC-III dataset is used to extract these different types to create three different datasets namely as structured, unstructured and combined dataset. Structured dataset consisted chart events, lab results, comorbidities and demographic information while unstructured dataset consisted discharge summaries from notes, and the combined dataset comprised both the structured and unstructured data.

Each admission is represented by min, max and average values from chart and lab events in structured data while a single note in unstructured data is represented by six different embeddings. The combined dataset is represented using equal number of features (50 from each dataset) from both datasets, for this, principal component analysis is used to reduce the dimensionality of unstructured data. The experimented algorithms included LR, SVM, XGBOOST, FFNN and RF. The author concluded that using Bag-Of-Words embedding on unstructured dataset, the Logistic Regression model had achieved best results by gaining AUROC score of 0.757 and recall of 0.682.

In 2022, Shi, K., et al. [36] investigated whether ML models can assess the likelihood of a 7-day ICU readmission upon discharge better than the existing benchmark, the Stability and Workload Index for Transfer (SWIFT) score. The data from Stanford Hospital (2009-2019) was used to train and validate the gradient boosting, random forest, support vector machine, and logistic regression models. The Beth Israel Deaconess Medical Centre (BIDMC) data (2008-2019) was used for external validation. Among the models mentioned above, Gradient Boosting had the best performance, with internal and external validation by obtaining F1-scores of 0.43 and 0.14 and AUROCs of 0.85 and 0.60, respectively.

In 2022, Pishgar, M., et al. [16] carried research where a process of mining/deep learning method is proposed for predicting an unscheduled 30-day return of heart failure patients in ICU. With the help of the DREAM (Decay Replay Mining) algorithm, time information obtained. To further improve the prediction accuracy, demographic data and severity ratings at the time of admission were merged with the time information and fed to a neural network (NN) model. Furthermore, a number of ML algorithms were created as the baseline models for the comparison of the findings. MIMIC-III dataset is used with having 3411 heart failure patients. In comparison to the existing literature available, proposed model yielded an AUROC of 0.930, the precision of 0.886, sensitivity of 0.805, accuracy of 0.841, and F-score of 0.800 which clearly showed better performance than the existing literature results. The proposed approach models the time-related variables by incorporating the patients' medical history and shows improved outcome.

In 2022, Orangi-Fard, N., A. Akhbardeh, and H. Sagreiya [37] employed text mining and ML on MIMIC-III to predict ICU 30-day readmission. Among the fifteen different types of data available in noteevents, they only used discharge summaries. The Bag-of-Words approach is used from NLP to build a Document-Term-Matrix with 3000 features. These features were used as the input features to different models like support vector machines with the radial basis function kernel (SVM-RBF), quadratic discriminant analysis (QDA), adaptive boosting (AdaBoost), Ridge Regression, and least absolute shrinkage and selection operator (LASSO) for performance comparison. Models were trained and validated using 4000 and 6000 patients respectively. Using full feature set, the AUROC curve was 0.71, 0.68, 0.65, 0.69, and 0.65 respectively for SVM-RBF, AdaBoost, QDA, LASSO, and Ridge Regression. Performance of models got increased using SVM-RBF model feature selection process which resulted in only 825 features or words. Upon using these selected features, the AUROC curve was 0.74, 0.71, 0.69, 0.67, and 0.70 respectively for SVM-RBF, Ridge Regression, AdaBoost, QDA, and LASSO. Summary of all the studies mentioned under this section in given is Table 2.2.

- **Deep Learning Models**

In 2018, Rafi, P., A. Pakbin, and S.K. Pentyala [38] suggested using Interpretable Mimic Learning, a Knowledge-Distillation technique, to forecast 30-day ICU readmissions. This architecture consisted of a teacher model (LSTM) and a student model (XGBoost) where teacher model learns complex features and transfers the knowledge to student model which makes final predictions. This method allows combining the accuracy and sequential learning of deep models with the interpretability of basic models, transferring deep model (LSTM and then DNN) knowledge (features) to simple and understandable models (XGBoost).

In 2018, Wang, H., et al. [39] done research where CNN is used for automatically extracting features from time series data, as well as categorical feature embedding, including demographics, hospitalization history, vital signs, and laboratory test. An MLP is given both statistical characteristics from feature embedding and learned features from CNN. To discourse the imbalance and skewness challenge, they trained MLP during prediction using a cost-sensitive formulation. Using two datasets from Barnes-Jewish Hospital, they validated their methodology. Using data from general hospital wards to predict 30-day readmissions, the proposed model's AUC was 0.70, which was significantly higher than all the baseline approaches.

Table 2. 2: Summary of Results using Conventional ML Models

Year	Author	Dataset	Technique	Accuracy (ACC %)	AUC (%)
2022	[36]	STARR	Gradient Boosting Model (GBM)		ROC=0.85
		MIMIC-IV			ROC=0.60
2022	[37]	MIMIC-III	SVM with the radial basis function kernel (RBF)		ROC=0.74
2022	[16]	MIMIC-III	Neural Network and using Decay Replay Mining (DREAM)	0.84	ROC=0.93
2021	[35]	MIMIC-III	Logistic Regression (LR)		ROC=0.76
2020	[34]	Nationwide Readmissions Database (NRD)	Artificial Neural Network (ANN)		ROC=0.72
2020	[33]	MIMIC-III	Random Forest (RF)	0.65	ROC=0.66
2019	[32]	MIMIC-III	NLP + Logistic Regression (LR)		ROC=0.75
2019	[29]	Geisinger Health System	Gradient Boosting Decision Tree (GBDT)		ROC=0.65
2019	[31]	GICU & MIMIC-III	Random Forest (RF) with extended feature set	0.84 - 0.85	ROC=0.86 - 0.88
2018	[28]	MIMIC-III	Extreme Gradient Boosting (XGBoost)		ROC=0.76
2018	[30]	University of Chicago dataset & MIMIC-III	Gradient Boosted Machine (GBM)		ROC=0.71 – 0.78

In 2019, Lin, Y.-W., et al. [40] predicted 30-days ICU readmission using RNN based LSTM+CNN, CNN+LSTM, LSTM and CNN model architectures. Researchers have also trained multiple conventional ML algorithms like Support Vector Machine, Logistic Regression and, Random Forest etc. for comparison purposes. MIMIC-III dataset was used to train and test all above mentioned algorithms having 35,334 patients with 48,393 ICU stays records after data preprocessing. Last 48 hours data before a patient is discharged or down-transfer is used for temporal features. 17 chart events features, 300 dimensional embeddings for ICD codes for diagnosis and 4 demographics features are the main extracted data features used here. For training conventional models, basic statistical and advanced features are extracted from temporal data like mean, slope and intercept computation etc. Author suggested RNN based LSTM+CNN architecture as the best model for prediction as it has performed well with AUROC of 0.791 and a sensitivity rate of 0.742.

In 2019, Ashfaq, A., et al. [41] predicted 30-day readmissions task for patients suffering from Congestive Heart Failure (CHF) to reduce readmissions. To predict unplanned readmission, they have taken over 7500 patients with CHF between 2012 and 2016 in Sweden. By using expert features and embeddings on clinical concepts, they have tested a cost-sensitive LSTM network. In a single framework, their study focused on three essential components of an EHR driven prediction model. By assessing each component's contribution, they demonstrated that the model outperforms in at least two evaluation metrics (AUC: 0.77; Cost: 22% of maximum feasible savings).

In 2020, Barbieri, S., et al. [42] presented different deep learning architectures for 30 days readmission risk prediction using interpretability of attention-based models on MIMIC-III dataset. Several deep learning architectures were trained, including those incorporating attention mechanisms, and medical concept embeddings with time-aware attention. Additionally, odds ratios were calculated for static variables to determine their association with an increased risk of readmission. Diagnoses, medications, procedures, and vital signs were ranked based on their contribution towards readmission risk. Amongst the architectures tested, a RNN, computed by neural ODEs, achieved the highest AUROC of 0.739 and an F1-Score of 0.372.

In 2020, Zhang, D., et al. [43] have performed three prediction tasks including in-hospital mortality, 30-day hospital readmission, and length of stay for MIMIC-III dataset. Here two multi-modal neural network architectures (Fusion-CNN and Fusion-LSTM) are proposed which combined the unstructured and structured data to develop patient representation learning. Structured data included temporal info (vital signs etc.) and static info (demography etc.) while unstructured data included the clinical notes by nurses or physicians. Depending on model's architecture, structured data is been modeled by CNN or LSTM networks while clinical notes are modeled by document embeddings. Hence a full patient representation is created by concatenating three representations from other modalities of data and fed as input to the model. On model's evaluation, it got evident that the proposed model performs better on combination of unstructured and structured data than other models which used either one kind of data only. For 30-day hospital readmission, Fusion-LSTM achieved better results than Fusion-CNN with AUROC score of 0.674 and AUPRC score of 0.079.

In 2022, Moazemi, S., et al. [15] did experiments with two models trained on MIMIC-III dataset (public) and validated on UKD dataset (private) for the patients who visited cardiovascular care units. The final dataset resulted in 11,513 patients, out of which 966 patients were labeled as positive and 10,547 were labeled as negative. Lab Values (Creatinine, Blood PH, Sodium, Potassium, Hematocrit, Bilirubin), Vital Signs (Body Temperature, ABP, Heart Rate, Oxygenation) and Patient Info (Age, Weight, LoS) are the main extracted data features used here. A special kind of RNN based LSTM model is proposed and trained on two datasets forming two models for further evaluation on test dataset. One is trained on all-time series data available before ICU discharged and the second one trained using only the 48 hours window before ICU discharge. Model with 48 hours window outperformed using RNN (LSTM) with AUROC 0.82.

In 2023, Kessler, S., et al. [14] researched on data samples which were taken from two cardiovascular ICUs (CCU and CSRU) and predicted 2-days ICU readmission. Researchers have also trained other 5 already known models Logistic Regression, Random Forest, ET, feed forward neural network (FNN), LSTM+CNN and one deep learning model (proposed LSTM model) have been trained and validated using an 80, 20 train test split strategy. MIMIC-III dataset was used to train and test all above mentioned algorithms having 12,797 ICU stays records after data preprocessing. Last 48 hours data before a patient is discharged or down-transfer is used for temporal features. Lab Values (Creatinine, Blood PH, Sodium, Potassium, Hematocrit, White blood cell counts, Bicarbonate, Bilirubin), Vital Signs (Body Temperature, ABP, Heart Rate, Oxygenation) and Patient Info (Age, Weight) are the main extracted data features used here. Author suggested proposed RNN based LSTM architecture as the best model for prediction as it has performed well with AUROC of 0.860 and AUCPR of 0.706.

Table 2. 3: Summary of Results using Deep Learning Models

Year	Author	Dataset	Technique	Accuracy (ACC %)	AUC (%)
2023	[44]	MIMIC-III	BERT		ROC=0.75 PRC=0.30
2023	[14]	MIMIC-III	RNN+LSTM		ROC=0.86 PRC=0.71
2022	[15]	UKD Dataset & MIMIC-III	RNN+LSTM		ROC=0.82 PRC=0.57
2020	[43]	MIMIC-III	Fusion-LSTM		ROC=0.67 PRC= 0.79
2020	[42]	MIMIC-III	RNN+ neural ordinary diff. equations with time-aware attention		ROC=0.74
2019	[41]	Halland Hospital	LSTM		ROC=0.77
2019	[40]	MIMIC-III	LSTM+CNN		ROC=0.79
2018	[39]	GHWs	DNN+MLP	0.89	ROC=0.70
		ORP		0.82	ROC=0.73
2018	[38]	MIMIC-III	Extreme Gradient Boosting (XGBoost)		ROC=0.71

In 2023, Sheetrit, E., M. Brief, and O. Elisha [44] have analyzed the difficulty to predict unplanned 30 days readmissions in ICUs and presented different experiments for the same task. By utilizing the two main models BERT and Gated Recurrent Network based bidirectional RNN network (BIRNN), they have assessed readmission for all four modalities of data from MIMIC-III dataset individually and then in combination. Standard architecture from Hugging Face used for BERT while the architecture of BIRNN has four main layers; the Embedding layer, two GRUs, an attention layer with a fully connected layer. The data modalities included static info like patient demographics, sequential info for diagnoses and prescriptions, unstructured clinical discharge notes, and multivariate time-series data, such as lab results and measurements etc. By assessing each data modality using the evaluation process, they concluded that BERT model trained on the Discharge notes outperformed others with AUROC score of 0.7522 and AUPRC score of 0.2988. Summary of all the studies mentioned under this section is given in Table 2.3.

Discussion

The review conducted in this chapter provides a summary of the prediction techniques applied on different public and private datasets from 2018 – 2023. From summary tables and 2.3, we see 11 studies have used conventional machine learning models while 9 studies have used deep learning models. If we compare performance, we can see that using conventional models, lowest ROC score reported is 0.60 and highest ROC score reported is 0.93 while using deep learning models, ROC score reported by almost all studies is around 0.75. Through this, we can analyze that although there are fewer articleswhich have used deep learning, yet their reported results are consistent compared to others'.

Figure 2.3 shows year wise number of publications included in our survey. As it is already mentioned that this survey includes articles published from 2018 to 2023, the maximum number of publications included in our survey are from year 2019. Figure 2.4 depicts technique wise number of publications. LSTM is very popular technique of RNN. Although fewer numbers of studies reported to be using deep learning techniques, 6 studies have used LSTM making it greatest reported 'used' technique.

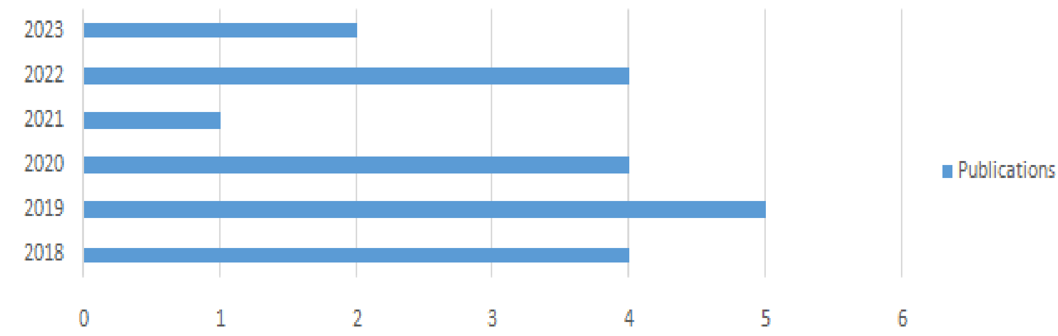

Figure 2. 3: Year Wise Number of Publications

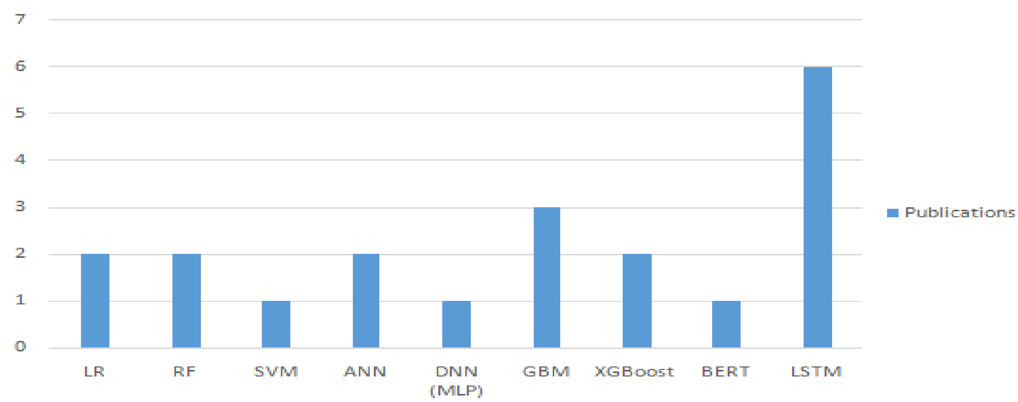

Figure 2. 4: Technique Wise Number of Publications

Table 2.4 demonstrates the summary of survey articles on the basis of core characteristics. The survey reveals that the Medical Information Mart for Intensive Care (MIMIC-III) is the most common database used by most of the researchers. The description of database has also been provided in chapter 3. Most of the studies have used time series data as almost all studies have used lab and chart measurements as their main features. 50% of the included studies have focused readmission within 30 days of patients' discharge. Although articles mentioned in survey are mostly those which haven't targeted any specific diagnosis yet there are considerable numbers of articles targeting specific disease like heart failure. This chapter provides a comprehensive analysis of different prediction models, which can help a practitioner choose his/her model for the underlying prediction problem.

Table 2. 4: Summary of Survey Articles on the Basis of Core Characteristics

Category	Sub Category	Count
Database	MIMIC-III	12
	Combined or Other Mixed	8
Approach	Conventional Machine Learning	13
	Deep Learning	7
Nature of data	Time Series	14
	Other	6
Diagnosis specific readmission	Yes	6
	No	14
Readmission after days discharge	30 Days	10
	2 Days	2
	NIL	8

Research Gap

The environment in ICU demands a significant staff-to-patient ratio and generates a substantial amount of data. The interpretation of real-time data and decision-making poses a terrible challenge for clinicians. The cost of intensive care is huge, which necessitates careful thought of when patients should be transferred to lower-level ward care. This decision is crucial to optimize the allocation of resources so, it is important to strike a balance in determining the appropriate timing for step-down, ensuring that patients receive optimal care while minimizing the likelihood of readmission.

From the literature survey, we see that limited research carried on readmission prediction task and the methods used were unable to attain good results. The articles included have also emphasized the usage of clinical notes and time series data as these are critical in decision making. Most of the studies have used traditional machine learning techniques. One drawback of traditional machine learning models is that patients' vectors are formed by aggregating patient features presented in observation window. Hence the temporality, which is important in healthcare situations, is ignored. So, we will move forward from here by choosing a technique that will keep temporality and will process the features sequentially.

Chapter 3

MATERIAL AND METHODOLOGY

This chapter is about the methods under taken and the material used to carry out this research. Material used will be elaborated in detail in data preparation section and remaining sections are about methodology explanation. These will be explained here under one by one. Flow diagram of proposed ICURP framework is also given in Figure 3.1 which helps to present the summary of this chapter.

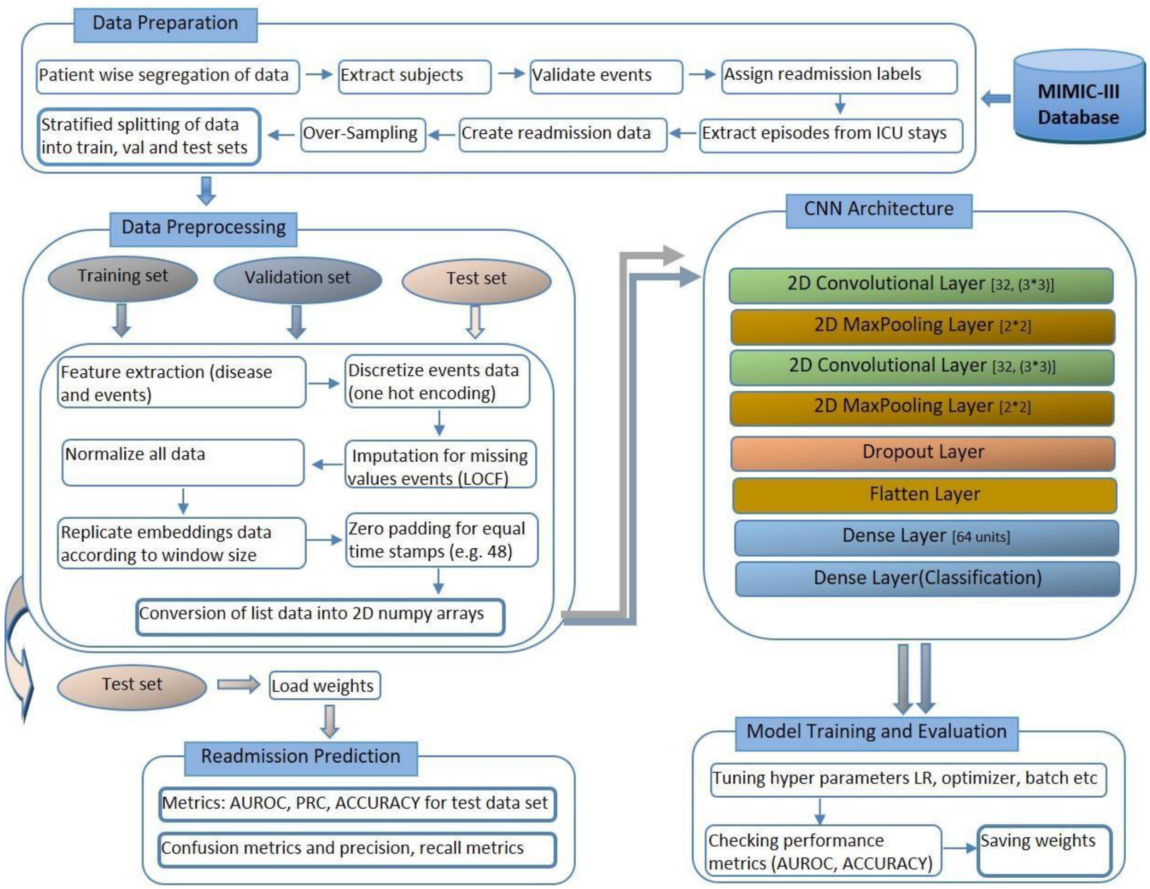

Figure 3. 1: Flow Diagram of Proposed ICURP Framework

Dataset construction will be explained first by explaining database archeology and data preparation steps. Data preprocessing steps will be explained second. Then the machine learning architecture will be explained in detail that will lead to the explanation of model training and evaluation stage. At the end, prediction process is explained.

Dataset Construction

The dataset for readmission prediction task is constructed from the MIMIC-III Critical Care Database which is maintained by the Massachusetts Institute of Technology (MIT) 's Laboratory for Computational Physiology [45]. MIMIC-III (Medical Information Mart for Intensive Care III) and eICU (electronic Intensive Care Unit) are the two publicly available datasets for this task; remaining datasets used in literature are private datasets. MIMIC-III contains 61,532 ICU stays with demographics, medications, vital signs, tests from laboratory and additional clinical data. Data included here can be categorized in two types as structured (static and temporal info) and unstructured (clinical notes) data. It has records of 46,620 patients of the Beth Israel Deaconess Medical Center between 2001 and 2012. Data preparation steps are followed from [14] and those patients are removed who are under the age of 18 and who died in the ICU. After applying major Data Preparation steps, we got 35,334 patients having 48,403 ICU stays which were used further in Data Preprocessing tasks. To build the dataset for ICU readmission, all chosen patients and their corresponding ICU stays records were classified as positive or negative cases while keeping 30 days movements (down transfer or up transfer) in mind.

Positive patient stays: Patients who can take advantage from early readmission prediction. In our dataset, among 48,403 ICU stays, we got 9,587 positive ICU stays. These are further categorized as:

i) Patients who returned to the ICU after being transferred to low level wards.

ii) Patients who died after transferred to low level ward.

iii) Patients who returned to the ICU within 30 days after being discharged.

iv) Patients who died within 30 days after being discharged.

Negative patient stays: Patients who do not need ICU readmission means being discharged or transferred from ICU and for next 30 days, they did not return and are still alive. In our dataset, among 48,403 ICU stays, we got 38,816 negative ICU stays.

MIMIC-III Database Archaeology

Table 3. 1: MIMIC-III Database Tables

Sr. No.	Table Name	Description
For defining and tracking patient stays		
1	ADMISSIONS	Every record of hospitalization for patient
2	CALLOUT	Information regarding ICU discharge
4	ICUSTAYS	Every record of ICU stays in the database
5	PATIENTS	Every record of patient in the database
6	SERVICES	The patient registered clinical service
7	TRANSFERS	Patients bed-to-bed movement within hospital
Contain data collected in the critical care unit		
8	CHARTEVENTS	Charted observations for patients
9	NOTEEVENTS	Nursing notes, ECG reports, and discharge summaries
10	OUTPUTEVENTS	Output information for patients
Contain data collected in the hospital record system		
11	DIAGNOSES_ICD	Hospital assigned diagnoses
12	DRGCODES	Diagnosis Related Groups (DRG)
13	LABEVENTS	Laboratory measurements records
14	PRESCRIPTIONS	Patients' medications records
15	PROCEDURES_ICD	Patients' procedures records
The following tables are dictionaries		
16	D_ICD_DIAGNOSES	Dictionary of ICD codes relating to diagnoses
17	D_ICD_PROCEDURES	Dictionary of ICD codes relating to procedures
18	D_ITEMS	Dictionary of ITEMIDs in the MIMIC database

This is a relational database comprising 26 tables of patient's records that stayed in the ICUs at Beth Israel Deaconess Medical Center. Database table names and their description are given under in Table 3.1. 16 tables have information on time-stamped events. To obtain the label of the itemid supplied in the chartevents table for this study, we have mostly used the chartevents table in conjunction with the connected d_item table. The mapping may be found in the *diagnoses_ icd* and *procedures_ icd* tables. International Classification of Diseases version 9 (ICD-9) codes are used to encode diseases. The majority of data is stored with a time stamp identifying the event's date and time (CHARTTIME and STORETIME, respectively). Dates are internally consistent for the same patient but inconsistent across patients as patients' data has been de-identified and all dates have been arbitrarily pushed into the future.

Data Preparation

This section explains in detail how MIMIC-III database is been processed so that it can be used further for readmission prediction task. For simplicity, data preparation tasks are performed on structured data. The statistics taken before data preparation step are given in Table 3.2.

Table 3. 2: MIMIC-III Database Statistics

Sr. No.	Description	Count
1	Records of patients	46,520
2	Records of admissions (visits)	58,976
3	Records of chart events (i.e., labs, tests, vital signs)	330,712,483
4	Records of diagnoses records	651,047
5	Records of procedures records	240,095

Data preparation steps are narrated below in the sequence order as they are executed. These are:

- **Patient wise segregation of data:** This step involves creating folders for each patient and keeping all other patient information like diagnoses, procedures, events and prescriptions etc. in these folders respectively.
- **Extract subject:**
    - Only those ICU stays are kept which have patients having age greater than 18 years.

- Only those records of Diagnoses, Procedures, Prescriptions and Events (charts, labs, and output etc.) are extracted which are related to ICU stays kept in previous step.

- **Validate events:** This is done to maintain the integrity of the records left.
- **Assign readmission labels:** In this step, labels are assigned to ICU stays by following the criteria as under;
  - 2: patients who died in ICU.
  - 1: positive; patients who are transferred back to ICU after being discharged/transferred to hospital ward or who died in less than 30 days after they are discharged.
  - 0: negative; patients who are not transferred back to ICU after being discharged/ transferred to hospital ward or who do not die in less than 30 days.
- **Extract episodes from ICU stay:** One patient can have multiple ICU stays. This step involves creating separate files for time series and non-time series data against each ICU stay. One ICU stay will have one episode file and that one episode would have;
  - Basic info data: This includes patient's demographic data.
  - Time series data: This includes patient's measurements in laboratory, output measurements and charted measurements.
- **Create readmission data:** This is the final step of data preparation which involves:
  - Removing un-labelled data.
  - Combining all ICU stays together in one separate file.
  - Putting time series files data at separate place.
  - Extracting meaningful features from time series data. These are those features which carry importance for the problem under discussion.
- **Over-Sampling:** From the last step, we got 48,403 ICU stays with 9,587 positive ICU stays and 38,816 negative ICU stays. Machine learning models may be trained on such data, but may not be able to reliably predict the minority class labels due to the imbalance in the label distribution [18]. In our scenario, we have less positive but large number of negative samples i.e., Positive Samples: 9,587, Negative Samples: 38,816, Total Samples: 48,403. Therefore, we have a very low 0.19 (19%) prevalence of readmission in our dataset. So, for adequate training and evaluation of ML models,

we must do oversampling before splitting this data into train, validation and test sets. From total samples, we have picked 9,587 positive ICU stays and replicated them twice and we have chosen the negative samples randomly in equal numbers as of positive samples. Now the data contains 38,348 samples with balanced labels and we moved further for creating data splits.

- **Splitting data into train, validation and test sets:** Data splitting is carried in 70%, 15% and 15% ratio for train, validation and test sets respectively. Five types of train, validation and test files sets with different samples are created; here we have used just one of them among all for our all experiments to ensure results consistency. Statistics related to distribution of samples among the three splits is described in Table 3.3.

Table 3. 3: Train, Validation and Test Data Split Statistics

Sr. No.	Data Split	Positive Samples	Negative Samples	Total Samples
1	Train Set	13422	13421	26843
2	Validation Set	2876	2876	5752
3	Test Set	2876	2877	5753

Data Preprocessing

After the MIMIC-III database being split into three separate dataset files for model training and testing, data is further preprocessed. The preprocessing pipeline consists of several chronological steps illustrated as a separate data preprocessing block in Figure 3.1. These preprocessing steps are executed for all data splits and the course of action is as follows: first, the two vital features (ICD-9 embeddings for disease diagnosis, chart measurements (events)) are obtained from the dataset. For the sake of comparison between new machine leaning models and conventional models, we will use time series nature of chart events as well as other basic and statistical features of the same chart events. Feature values are discretized then missing data is filled and normalization is performed. These steps are followed by data replication and zero padding. Finally, the list format data is converted into 2D numpy arrays before being passed to the model. The mechanism is explained in more detail here under:

Feature Extraction

It is the major step of data preprocessing phase so it will be explained in more detail likewise. We have chosen the time series window being used and extracted the features for the ICU readmission prediction task. For modeling of the time series ICU records, we use the last 24-hour, last 48-hour and last 72-hour measurements data of each ICU stay. The last hour values before the patient is discharged or down transferred to ward, are found to be the most useful data for prediction of readmission [46]. First, chart events features are extracted from health care notes provided by physicians and nurses. The experts make their observations while examining the patients' physiological conditions and these opinions are represented in chart events [20]. Second, the patient's diseases diagnosis measurements are extracted. These diseases diagnosis features are strongly associated with ICU readmission [46].

Chart events (Temporal Information): These events are the measurements made while examining patients from laboratory results, output and chart events measurements (vital signs). We have combined these three types in a single type and named it as chart events. Table 3.4 shows in detail the 17 temporal features with their dimensions besides showing their normal values in the humans. It is evident from table that chart events have 59 dimensions in total. These features are time series in nature, as these can be the measurements made on hourly basis or can have more than one measurement in an hour. Measuring window is 24-hour, 48-hour and 72-hour long, and occupies both types of features e.g., numerical (diastolic blood pressure) and categorical (capillary refill rate).

ICD-9 embeddings: One of the most significant factors linked to readmissions is chronic illness [46]. ICD codes symbolize specific diagnoses assigned to patients during their medical visits. These codes provide information about various diseases and conditions such as diabetes mellitus without complications (code 250.00) and cardiovascular disease (code 429.2). With a vast array of approximately 18,000 ICD-9 codes available, a simple approach to convert them into features is to create a binary feature vector consisting of 18,000 elements. Each element in the vector shows whether the patient has the corresponding disease or not. However, employing this approach would result in a sparse and high-dimensional feature vector. Here, we used the method described in [22] which suggests using the pre-trained embeddings to address the data scarcity. A pre-trained 300-dimensional embedding is computed against each ICD-9 code and utilized instead of ICD-9 code for disease. By avoiding a sparse representation and using the relationship information between various diseases, employing a lower dimensional embedding of the ICD-9 is advantageous for the model training process. In order to build the feature for a patient with several diagnoses' conditions, we only added the embeddings of each condition.

Table 3. 4: 17 Chart Events (Temporal) Features

Sr. No.	Chart Events	Possible Values	Dimensions	Normal Values
1	Glasgow coma scale eye opening	'To Pain', '3 To speech', '1 No Response', '4 Spontaneously', 'None', 'To Speech', 'Spontaneously', '2 To pain'	8	4 Spon-taneously
2	Glasgow coma scale verbal response	'1 No Response', 'No Response', 'Confused', 'Inappropriate Words', 'Oriented', 'No Response-ETT', '5 Oriented', 'Incomprehensible sounds', '1.0 ET/Trach', '4 Confused', '2 Incomp sounds', '3 Inapprop words'	12	5 Oriented
3	Glasgow coma scale motor response	'1 No Response', '3 Abnorm flexion', 'Abnormal extension', 'No response', '4 Flex-withdraws', 'Localizes Pain', 'Flex-withdraws', 'Obeys Commands', 'Abnormal Flexion', '6 Obeys Commands', '5 Localizes Pain', '2 Abnorm xtensn'	12	6 Obeys Commands
4	Glasgow coma scale total	'11', '10', '13', '12', '15', '14', '3', '5', '4', '7', '6', '9', '8'	13	15
5	Capillary refill rate	'0.0', '1.0'	2	0.0
6	Diastolic blood pressure		1	59.0
7	Systolic blood pressure		1	118.0
8	Mean blood		1	77.0

	pressure			
9	Heart Rate		1	86.0
10	Glucose		1	128.0
11	Fraction inspired oxygen		1	0.21
12	Oxygen saturation		1	98.0
13	Respiratory rate		1	19.0
14	Body Temperature		1	36.6
15	pH		1	7.4
16	Weight		1	81
17	Height		1	170.0

Data Discretization

This step is majorly performed on the time series data obtained for chart events. Since we already know, chart events have 17 types of features. Among these 17 features, 5 are categorical and remaining 12 are numerical features. Numerical features don't need any discretization since numerical values are well handled by machine learning models. While categorical data must be converted to numerical data through any means to be used by models for further processing. Many techniques are proposed in literature, but here we have used one-hot encoding. 'One Hot Encoding' is where labels are encoded in a way that each label gets equal weight. So, after conversion/discretization, we got 59 dimensions from the chart events which are surely the increased number. The same step also involves creating a binary mask for each feature (total 17) which indicates the existence of record for each feature. After performing all steps here, we got chart events with 76 features.

Handling Missing Data

It is not always possible that every feature has recorded measurements for every patient. The input to the models must have a constant size, therefore missing features must be restored or data from ICU stays that lack certain features cannot be included, leaving the dataset with insufficient data. So, handling most of the missing time series data, we need to use some well efficient techniques. Many techniques, in literature, are used to handle such discrepancies in data like deletion or imputation. While deleting records with missing values may cost us to lose important data, we have used Last-Observation-Carried-Forward (LOCF) method for missing values imputation. The last existing value is carried forward and imputed to fill the missing value.

Data Normalization

Normalization, a data preprocessing step, aims at scaling and transforming the data to ensure that all features have similar distributions. This helps to ensure the algorithm's learning efficacy and prevents certain features from dominating others. Here normalization is applied in two ways; on one hand, it unifies the features data which might have been stored with different measuring units. One the other hand, all features are normalized using the z-score standardization method which brings features values to have a mean of 0 and a standard deviation of 1. This equation is defined as:

$$\hat{s}_i = \frac{s_i - \mu_x}{\sigma_x} \tag{3.1}$$

where $\hat{s}_i$ is the normalized value, μ_x is the mean of the feature over the whole dataset x, and σ_x is the standard deviation over x.

Data Replication

This preprocessing step is very specific to this research. One ICU stay sample has one disease embedding vector while there are many chart events records against one ICU stay ID. In this step, disease embeddings records are replicated to as many times as hourly time series data (measurements) then concatenated. This step makes all data look similar so that it can be processed in temporal manner further. This process can be viewed in Figure 3.2.

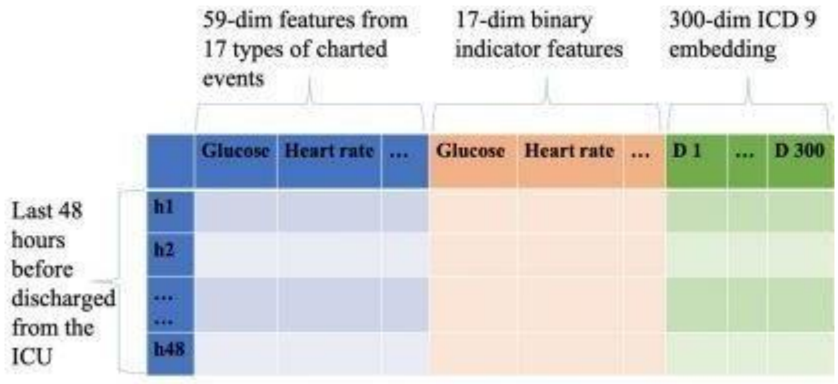

Figure 3. 2: Structure of Features after Data Replication

Zero padding for Timestamps Equalization

The input feature vectors must all be of the same size before being analyzed by the deep learning algorithms which we are using further in our experiments. Almost half of the ICU stay samples in our dataset did not have the 48 hours long ICU stay, consequently no 48 hours measurements. So, to consider 48-hour window size for experimentation, we have to pad the missing measurements with zero values. Table 3.5 provides the period length wise distribution of data samples. Number of padded rows depends on the window size we are using.

Table 3. 5: Period Length wise Data Distribution

Sr. No.	Length of Stay	No of Samples
1	<=24	6978
2	<=48	22510
3	<=72	31307

Conversion of List Data into 2D numpy Arrays

LSTMs are specifically designed for sequential data and capture long-term dependencies and CNNs are commonly used in image processing due to their ability to capture spatial patterns. In the case of time series, CNNs process data in a different way compared to traditional sequential processing. They use 1D or 2D convolutions to extract features across the sequential data, identifying patterns and dependencies within certain window sizes. Here we have applied 2D convolutions so, we have converted the data into 2D matrix. Time stamps represented the rows and feature dimensions represented the columns.

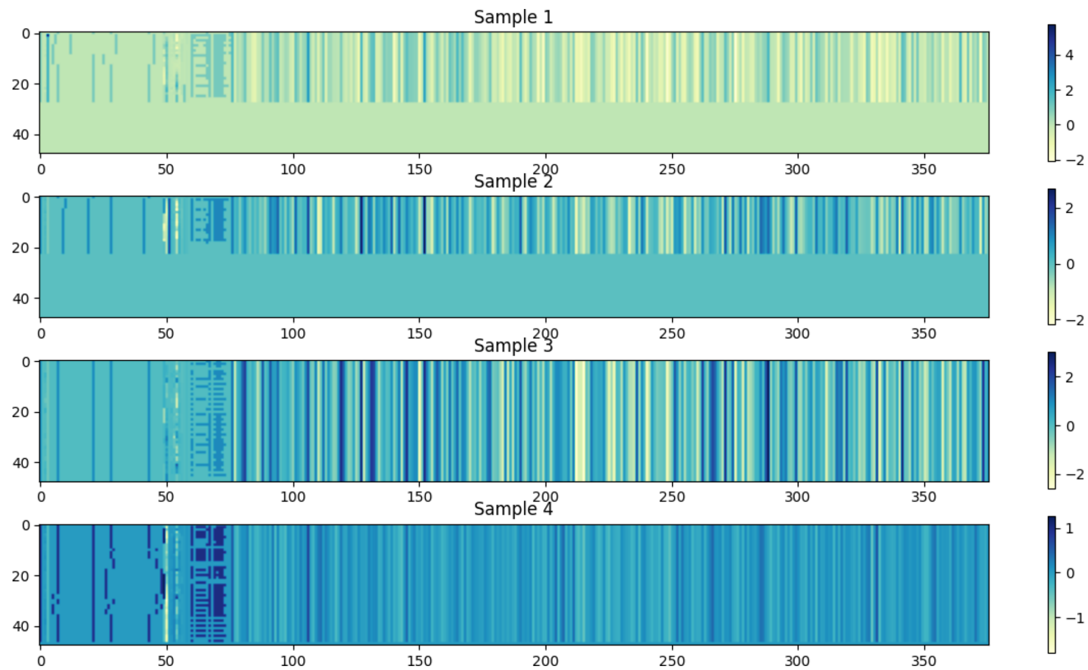

Figure 3. 3: 2D Visualization of Negative Samples

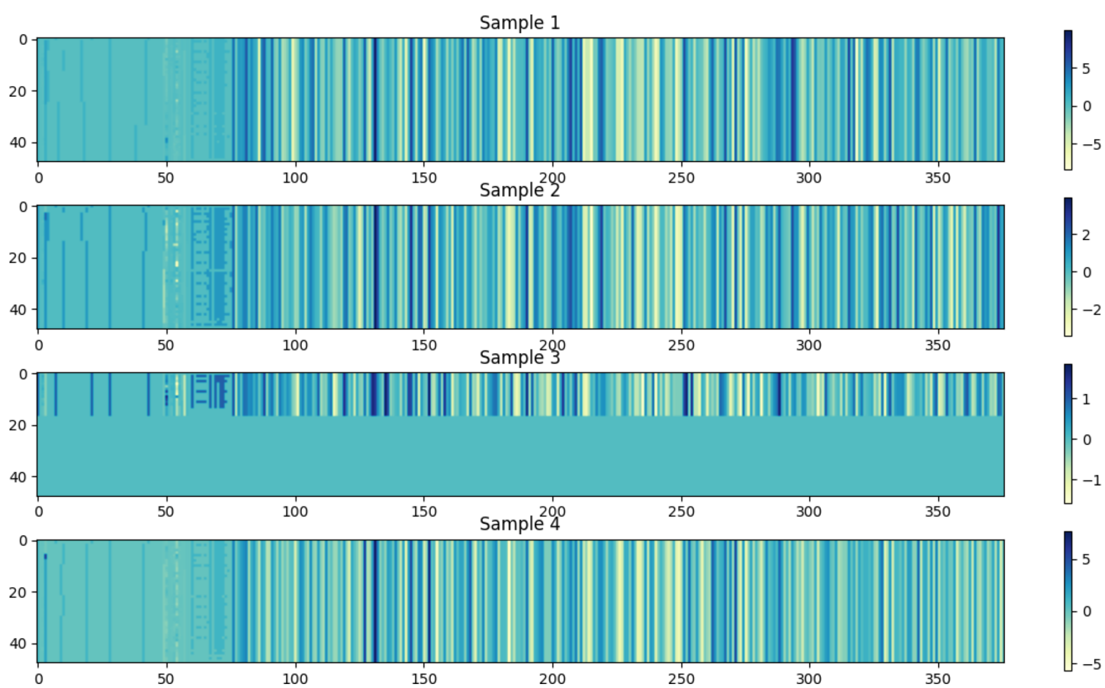

Figure 3. 4: 2D Visualization of Positive Samples

CNN Architecture

Artificial neural networks (ANNs) are fundamental components within deep learning methodologies. Among these networks, Recurrent Neural Networks (RNNs) specifically process sequential or time-oriented data. They find applicability in various fields such as NLP, language translation, speech recognition, and image captioning. On the other hand, the Convolutional Neural Network (CNN) is an expanded version of the ANN, adept at extracting crucial information from both time series and image-based data, making it particularly valuable for pattern recognition tasks [47]. CNNs utilize principles of linear algebra, like matrix multiplication, to identify patterns, and are capable of classifying not only images but also audio and signal data. The CNN's architecture, often referred to as covnets, comprises layers that sequentially transform one volume of data into another through differentiable functions. These layers are explained here under:

- **Input Layers:** Input Layers serve as the initial point where data is fed into the model. In CNNs, the input is typically a 2D matrix. This layer contains the raw input data structured with parameters like width, height, and depth.
- **Convolutional Layers:** Convolutional Layers are fundamental components within a CNN responsible for the major part of the network's computational processes. These layers extract features from the input dataset by applying learnable filters, or kernels, to the input data. Kernels, typically smaller matrices like 2×2, 3×3, or 5×5 in size, move across the input, calculating the dot product between the kernel weights and the corresponding input values. The result is known as feature maps. The output volume's size can be determined using a formula based on the input size (W x W x D), the number of kernels (Dout), the kernel's spatial size (F), the stride (S), and the amount of padding (P), then the size of output can be determined by the following formula:

$$W_{out} = \frac{W - F + 2P}{S} + 1 \tag{3.2}$$

- **Activation Layer:** The Activation Layer introduces nonlinearity to the network by implementing an activation function on the output of the previous layer. It applies an element-wise activation function, such as RELU (max (0, x)), Tanh, Leaky RELU, Sigmoid, etc.

- **Pooling layer:** This layer is intermittently integrated into convolutional networks to decrease the volume's size, enabling faster computations, reduced memory usage, and preventing overfitting. Max pooling and average pooling are two frequently used types. Using an activation map of size W x W x D, a pooling kernel with a spatial size F, and a specific stride S, the output volume's size can be calculated using a particular formula:

$$W_{out} = \frac{W - F}{S} + 1 \tag{3.3}$$

- **Flattening:** After passing through the convolutional and pooling layers, the resulting feature maps are flattened into a one-dimensional vector. This step allows them to be fed into fully connected layers for classification or regression purposes.
- **Fully Connected Layers:** Fully Connected Layers establish complete connectivity between neurons in the preceding and subsequent layers. This connectivity allows standard computations through matrix multiplication followed by bias addition. The role of the FC layer is to facilitate the mapping of representations between the input and output.
- **Output Layer:** The output generated by the fully connected layers is directed into a logistic function, such as sigmoid or softmax. This step is crucial for classification tasks as it transforms the output of each class into the respective probability scores for those classes.

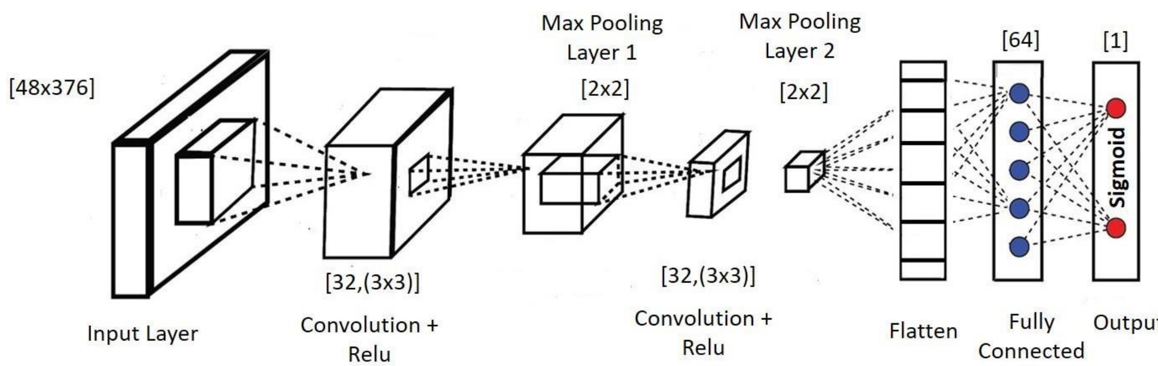

Figure 3. 5: CNN Architecture

The CNN architecture used in this research is shown in Figure 3.3. It has multiple layers; the used CNN architecture is explained here under:

- keras.Sequential: This creates a linear stack of layers for building the neural network. The layers will be added sequentially.
- layers.Conv2D(32, (3, 3), activation='relu', input_shape=(48, 376, 1)): This line creates the first convolutional layer. It includes 32 filters/kernels of size 3x3. The activation function used is Rectified Linear Unit (ReLU). The input shape for this layer is (48, 376, 1), which represents a 2D matrix with a height of 48 rows, width of 376 columns, and a single channel.
- layers.MaxPooling2D((2, 2)): This layer performs max pooling, which reduces the dimensions of the previous layer by taking the maximum value within each 2x2 window. This helps in reducing computation and controlling overfitting.
- layers.Conv2D(32, (3, 3), activation='relu'): The second convolutional layer with 32 filters of size 3x3 and ReLU activation. This layer learns more complex features as it goes deeper into the network.
- layers.MaxPooling2D((2, 2)): Another max pooling layer to further reduce dimensions.
- layers.Dropout(0.4): Dropout is a regularization technique that helps prevent overfitting by randomly setting a fraction (in this case, 40%) of input units to 0 at each update during training, which helps in creating robust networks.
- layers.Flatten(): This layer flattens the 2D output from the previous layer into a 1D array, which is necessary before passing the data to the densely connected layers.
- layers.Dense(64, activation='relu'): A fully connected (dense) layer with 64 neurons and ReLU activation function. It learns patterns from the features extracted by the convolutional layers.
- layers.Dense(1, activation='sigmoid', name="class"): The final dense layer with a single neuron, using a sigmoid activation function. This layer is for binary classification tasks as it produces a probability output between 0 and 1, representing the likelihood of the input belonging to 1 class. The layer is named "class".

Model Training and Evaluation

This step explains the hyper parameters used for training CNN model and is the basis for saving the trained model weights. Model is trained using training set and hyper parameters are tuned while validating the performance on validation set.

Tuning Hyper Parameters

- **loss = binary_crossentropy**: Sets the loss to 'binary cross entropy' since the problem under consideration is binary classification problem.
- **optimizer = Adam**: Sets the optimizer to 'Adam' which helps to optimize the weights while calculating gradients. Adam is an addition of the stochastic gradient descent (SGD) algorithm that incorporates adaptive learning rates and momentum.
- **metrics = accuracy**: The metrics parameter is used to specify one or more performance metrics that we want to track and evaluate during the training and validation of our model. In this case, we specified 'accuracy' as our required metric.
- **learning_rate = 0.001**: Sets learning rate which indicates the step size at which the model parameters will be updated during the optimization process.
- **batch_size = 128**: Sets the batch size to 128, indicating the number of samples processed before updating the model's weights during training.
- **epochs = 120**: Models is trained till 120 epochs.

Checking Performance Metric and Saving Weights

One type of performance metric is monitored i.e., validation accuracy. Model is trained using 120 epochs and while training, validation accuracy is monitored. Those model weights are saved which have improved validation accuracy.

Readmission Prediction

This is the final step of our proposed methodology. Weights with the highest validation accuracy are loaded and ICURP framework's performance is checked on unseen data (test set). Test set have to go through all data preprocessing steps which are being opted for training and validation sets. Once the test data is preprocessed, it is used for prediction. Confusion matrix, precision, recall, Accuracy, ROC, and PRC are monitored.

Summary

MIMIC-III database is huge database. Data Preparation steps mentioned above are used just to formulate the data according to the machine learning problem. Multiple Data Preprocessing steps are taken to extract the needed features and bring it into a form so that machine learning algorithms can perform training easily. The major part comes here which is about the machine learning model and model's architecture and for that we have chosen CNN. Number of layers, their types and units in each layer are decided. Then model is trained on training dataset and hyper parameters are tuned while seeing performance on validation dataset. Weights are saved while training and performance is evaluated using test dataset. All these steps are performed in the order in which they are mentioned here. Results of experiments with different machine learning models on varying window size are given in next chapter. Next chapter also elaborates the features modeling process used for different models experimented here in this research in detail.

Chapter 4

RESULTS AND DISCUSSION

This chapter is about the experimental results and the detail discussion of the obtained results explained here in detail. First section visually shows about the feature engineering strategy for model training while second section describes the test datasets. Third section mentions the performance metrics used to evaluate the experiments and the fourth section describes the detail of experiments. Last section discusses the above experimental results through comparison.

Training Machine Learning Models

Machine learning models are used to learn hidden patterns in data and make decisions on unseen data. For this research, many conventional machine learning and deep learning algorithms have been experimented. Conventional algorithms included Support Vector Machine, Logistic Regression, K Nearest Neighbor and Naïve Bayes while deep leaning algorithms included CNN, RNN based LSTM and GRU.

After following multiple processes in Data Preparation step, we got three dataset files for train, validation and test set. These dataset files have to go through multiple Data Preprocessing steps. Data Preprocessing phase have feature extraction, Data Discretization and 2D data conversion as their major steps. Two types of features are extracted and used to train all experimented models. These features are diagnosis features and chart event features. After data discretization process and using the ICD-9 embeddings for diseases diagnosis, we obtained 376 total features (ICD-9 Diagnosis Embeddings features: 300 & Chart Events features: 76).

For training conventional machine learning and deep learning models, features are modeled differently. Figure 4.1 shows the difference between two approaches. Here for training conventional models, temporal features are aggregated while for training deep learning models, static features are replicated. Through the feature engineering process adopted by conventional ML models, basic as well as advanced statistical features like coefficients, intercept are computed from numerical chart events. Binary mask and average occurrence of category is also computed and used for training these conventional models.

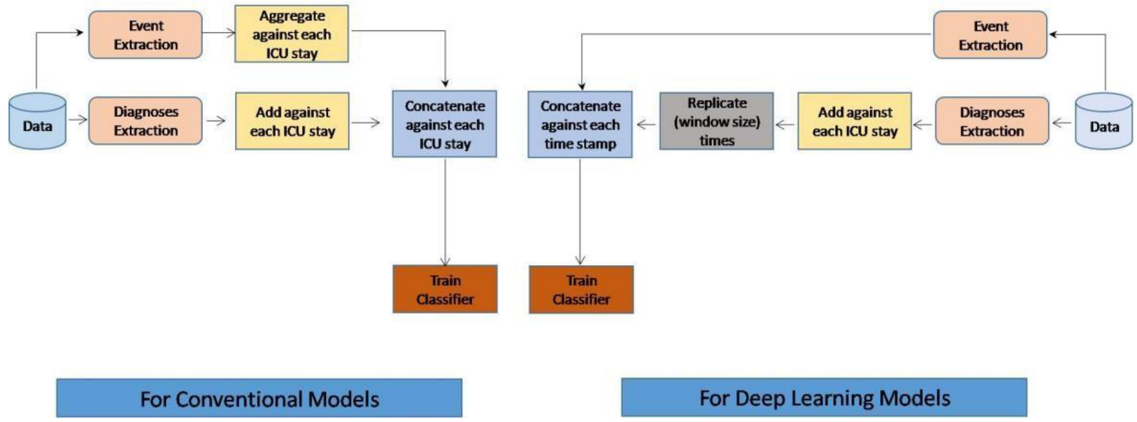

Figure 4. 1: Features Modeling for Training

Test Datasets

For evaluation, two unseen datasets from MIMIC-III are used. Dataset A contains the unseen test data created after over-sampling and stratified splitting of whole data while Dataset B contains data which haven't gone through over-sampling and stratified splitting techniques. Number of samples in A and B are 5753, 4797 respectively. Lin, Y.-W., et al. [40] have also used the same dataset B for evaluation in his research. Table 4.1 shows the data split statistics.

Table 4. 1: Test Data Split Statistics

Sr. No.	Dataset	Positive Samples	Negative Samples	Total Samples
1	Dataset A	2876	2877	5753
2	Dataset B	928	3869	4797

Performance Measures

Performance metrics are the criteria used to assess the performance of a classifier. They represent quantitative measures of how accurately the classifier predicted or classified the class variable. The building blocks that are used to compute many metrics are:

True positives (TP): TP refers to the positive tuples labeled positive by the classifier.
True negatives (TN): TN refers to the negative tuples labeled negative by the classifier.

False positives (FP): FP refers to the negative tuples labeled positive by the classifier (incorrect classification; negatives labeled as positives).

False negatives (FN): FN refers to the positive tuples labeled negative by the classifier (incorrect classification; positives labeled as negatives).

These terms are summarized in the confusion matrix shown in Table 4.2 which is a tool to analyze how well a classifier can classify the records. The evaluation metrics used to evaluate our proposed methodology and other classifiers are listed in Table 4.3. To calculate ROC, two parameters are required: true positive rate (TPR) and false positive rate (FPR), while for calculating PRC, precision and recall is required.

Table 4. 2: Confusion Matrix

Actual Class	Predicted Class	
	No	**Yes**
No	TN	FP
Yes	FN	TP

Table 4. 3: Performance Measures

Sr. No.	Performance Metric	Description	Formula
1	AUC ROC	Area under the receiver operating characteristic curve	
2	AUC PRC	Area under the precision recall curve	
3	Accuracy (ACC)	Percentage of tuples that are correctly classified by classifier	$ACC = \frac{TP + TN}{TP + TN + FP + FN}$
4	Precision	Percentage of positive predicted tuples out of all positive predicted	$Precision = \frac{TP}{TP + FP}$
5	Recall / Sensitivity (SEN) (TPR)	Percentage of actual positive tuples that are correctly classified as positive	$SEN = TPR = \frac{TP}{TP + FN}$
6	(FPR)	Percentage of negative tuples that are incorrectly classified as positive	$FPR = \frac{FP}{FP + TN}$

In our scenario, TP is the no. of patients who got readmitted in ICU after discharge/transfer and are correctly classified as patients readmitted. FP is the no. of patients who didn't get readmitted and wrongly classified as the patients who got readmitted. TN is the no. of patients who didn't get readmitted in ICU after discharge and are correctly classified as patients who didn't get readmitted and FN is the no. of patients who got readmitted and wrongly classified as the patients who didn't get readmitted.

Experimental Study

In this section, four experiments have been demonstrated. First subsection shows the experimental results using four conventional models. Second subsection demonstrates the experiments using RNN based LSTM and GRU models and third subsection presents experimental results using the ICURP framework's-based CNN model. All the experimental results are displayed one by one below in their respective section. The experiments were performed using training, validation and test datasets and the scores obtained on test sets are listed in tables and plotted as graphs as well.

For deep learning models, three further experiments are performed with widow size variations in feature set. Three models are trained using last 24-hour, 48-hour and 72-hour ICU time series data. Hence, the impact of varying window size is analyzed in different experiments. Important aspect to be considered here is that for all deep learning models, the chosen features and the experimental setup kept the same. Two test datasets A and B are used to get the evaluation values and these values are reported in result tables. While training deep learning models, validation accuracy is monitored, and weights with best validation accuracy are saved. The weights with highest validation accuracy are loaded and test dataset performance is evaluated and reported in tables here.

Major performance metrics used for analysis are: accuracy, area under the receiver operating characteristic curve (AUC of ROC) and area under the precision-recall curve (AUC of PRC). Among all five metrics i.e., ROC, PRC, Precision, Recall and Accuracy, if ROC, PRC and Accuracy have improved results compared to others then we have declared that method as our benchmark method. The receiver operating characteristic (ROC) curve illustrates the diagnostic capability of a binary classifier by plotting the TPR versus FPR at different cut-off or threshold points. The more the curve is inclined towards the top-left corner, the better the classifier is. The precision-recall curve (PRC) assesses the classifier's precision and recall trade-off. The top-right corner represents the optimal performance for Precision-Recall curve, with high precision and high recall simultaneously.

Experimentation with Conventional ML Models

This is the first experiment conducted in this research. Here four machine learning classifiers are trained and evaluated. These classifiers include Support Vector Machine (SVM), Logistic Regression (LR), K Nearest Neighbor (KNN) and Naïve Bayes (NB). LR is trained using 'l2' regularization having strong inverse regularization value. KNN is trained using 7 nearest neighbors. The dataset comprised of last 48-hour ICU data and evaluated using unseen dataset A. The result statistics of all classifiers are shown in Table 4.4. From the statistics, it is evident that SVM outperformed all other classifiers with ROC score 0.789, PRC score 0.776 and accuracy score 0.719 which are the highest among all other classifiers so, we are giving it priority. After SVM, LR can be ranked on second number. Its ROC and PRC scores are high with 0.774 and 0.765 values respectively. It has accuracy score of 0.702 which is higher than NB and KNN accuracy scores. Plot of ROCs for all conventional classifiers are shown in Figure 4.2.

Table 4. 4: Conventional ML Models Performance on Dataset A

Classifier	Accuracy	Precision	Recall	PRC	ROC
SVM	**0.719**	**0.722**	**0.713**	**0.776**	**0.789**
<u>LR</u>	<u>0.702</u>	<u>0.718</u>	<u>0.666</u>	<u>0.765</u>	<u>0.774</u>
KNN	0.687	0.680	0.706	0.741	0.756
NB	0.642	0.708	0.483	0.733	0.714

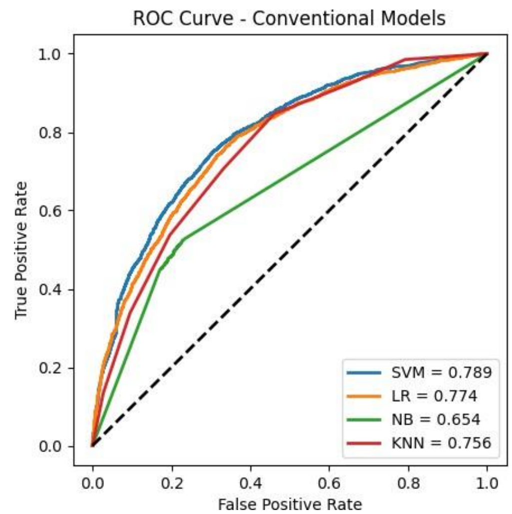

Figure 4. 2: Conventional Models ROC Curve

Experimentation using RNN based Models

Recurrent neural networks (RNN) are used most commonly for processing sequential and time series data [48]. Long-Short-Term-Memory Networks (LSTM) and Gated Recurrent Unit (GRU) are types of Recurrent Neural Network (RNN) introduced to handle the short-term memory problem of RNN. GRU was introduced by Cho et al. in 2014 as a simpler substitute to LSTM networks and it is generally well-suited for capturing short-term dependencies and patterns within sequences. For longer sequences, other architectures may be more appropriate. The basic difference between LSTM and GRU is their gating mechanism and number of gates.

- **Experimentation using LSTM**

This is the second experimentation technique. Hochreiter and Schmidhuber introduced LSTM as improvement over traditional RNNs to overcome their limitations. By incorporating additional interactions per module, LSTMs can learn long-term dependencies and retain information for extended periods effectively [49]. Figure 4.3 visually depicts the structure of LSTM model used here in our research. The model incorporates a LSTM layer with 16 units, which is a type of RNN layer suitable for sequential data. A dropout layer with a dropout rate of 0.4 is added for regularization, followed by a dense output layer with one unit and a sigmoid activation function for binary classification. The model is compiled till 120 epochs using the Adam optimizer with a learning rate of 0.001, binary cross-entropy loss, and accuracy as the evaluation metric. Additionally, the code sets up callbacks for model checkpointing, saving the best weights based on validation accuracy.

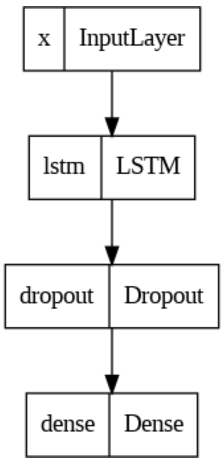

Figure 4. 3: LSTM Model Structure

Three alternative LSTM models are trained by changing window size and evaluated using dataset A. The evaluation results show that by using last 24-hour ICU data, the achieved accuracy, PRC and ROC scores are 0.757, 0.808 and 0.827 respectively. Using last 48-hour ICU data, accuracy, PRC and ROC scores got improved as compared to last 24-hour. It has achieved the accuracy, PRC and ROC scores of 0.775, 0.815 and 0.842 respectively. Third evaluation results are obtained by using last 72-hour ICU data. We see performance did not improve as compared to last 48-hour feature set as the accuracy, PRC and ROC scores are 0.765, 0.794 and 0.820 respectively. The result statistics for three experiments are presented in detail in Table 4.5. Figure 4.4 visually compares the ROCs and PRCs for the three experiments which are performed using LSTM.

Table 4. 5: LSTM Performance on Dataset A

Window Size	Accuracy	Precision	Recall	PRC	ROC
24-h	0.757	0.738	0.796	0.808	0.827
48-h	**0.775**	**0.764**	**0.793**	**0.815**	**0.842**
72-h	0.765	0.749	0.799	0.794	0.820

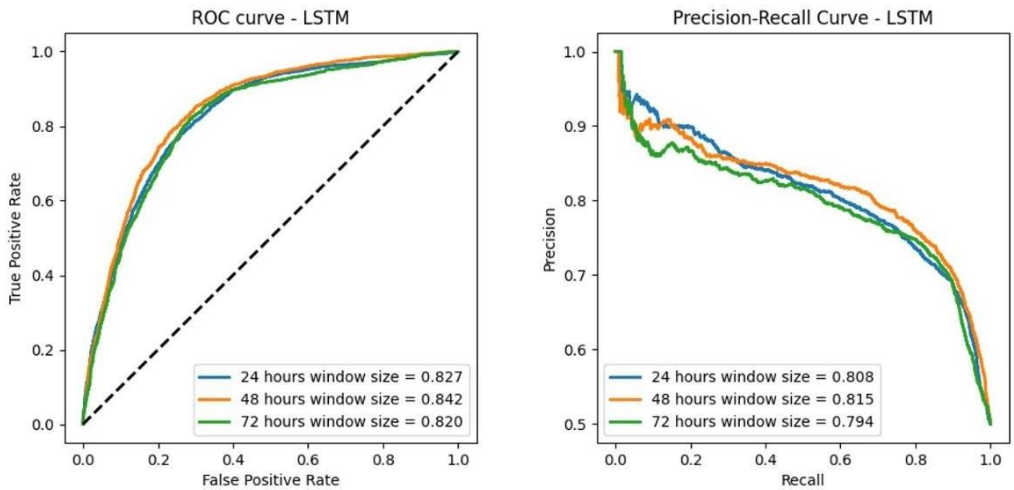

Figure 4. 4: LSTM - ROC & PR Curves

- **Experimentation using GRU**

This is the third experimentation technique. GRU has a simplified version of the LSTM cell to address the complexities while achieving improved network performance and faster training time. GRUs and LSTMs operate similarly, but GRU merges the hidden state and cell state into a single state, effectively reducing the total number of gates to half of what LSTM requires [50]. Figure 4.5 visually depicts the structure of GRU model used here in our research. The model incorporates a GRU layer with 16 units followed by a dropout layer with a dropout rate of 0.4. Finally, a dense output layer added with one unit and a sigmoid activation function for binary classification. The model is compiled till 120 epochs using the Adam optimizer with a learning rate of 0.001, binary cross-entropy loss, and accuracy as the evaluation metric. The code sets up callbacks for model checkpointing, saving the best weights based on validation accuracy.

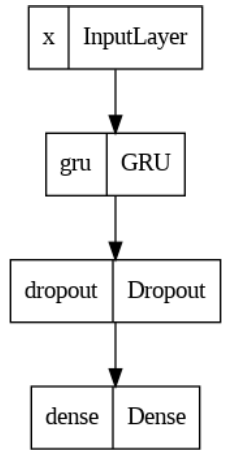

Figure 4. 5: GRU Model Structure

Three alternative GRU models are trained by changing window size and evaluated using dataset A. The evaluation results show that by using last 24-hour ICU data, the achieved accuracy, PRC and ROC scores are 0. 748, 0.799 and 0.821 respectively. Using last 48-hour ICU data, accuracy and ROC scores got improved but we see decline in PRC score as compared to last 24-hour. It has achieved the accuracy, PRC and ROC scores of 0.772, 0.788 and 0.826 respectively. Third evaluation results are obtained by using last 72-hour ICU data. We see performance did not improve as compared to previous two experiments. The result statistics for three experiments are presented in detail in Table 4.6. Figure 4.6 visually compares the ROCs and PRCs for the three experiments which are performed using GRU.

Table 4. 6: GRU Performance on Dataset A

Window Size	**Accuracy**	**Precision**	**Recall**	**PRC**	**ROC**
24-h	0.748	0.727	0.792	0.799	0.821
48-h	**0.772**	**0.749**	**0.817**	**0.788**	**0.826**
72-h	0.749	0.727	0.799	0.775	0.807

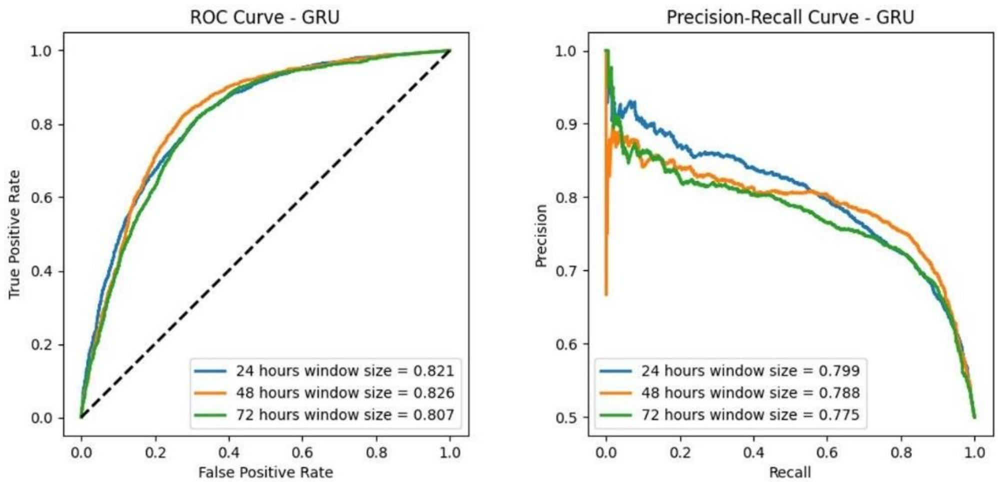

Figure 4. 6: GRU - ROC & PR Curves

Experimentation using ICURP Framework

CNN is an expanded version of the ANN, adept at extracting crucial information from both time series and image-based data, making it particularly valuable for pattern recognition tasks. Figure 4.7 visually depicts the structure of CNN model used here in our research. The model consists of two convolutional layers with 32 filters each, using a ReLU activation function. Max pooling layers with a (2,2) pool size are applied after each convolutional layer. A dropout layer with a dropout rate of 0.4 is included. The flattened output is connected to a dense layer with 64 units and a ReLU activation function, followed by a final dense layer with one unit and a sigmoid activation function, indicating binary classification. The model is compiled till 120 epochs using the Adam optimizer with a learning rate of 0.001, and binary cross-entropy loss function. Callbacks are set up for model checkpointing, saving the best weights based on validation accuracy. Using the methods suggested by ICURP framework, three alternative CNNs models are trained and validated using 24-hour, 48-hours and 72-hour window size on ICU data. For evaluation, two unseen datasets A and B are used. Evaluation of results using dataset A are presented in Table 4.7 and the results of evaluation on dataset B are shown in Table 4.8. Using dataset A, all the three models performed well and model trained using 48-hour window outperformed others with accuracy, PRC and ROC score of 0.816, 0.866 and 0.882. Figure 4.8 visually compares the ROC and PRC of the three experiments. Using dataset B, all the three models showed improved performance than on dataset A and model trained using 72-hour window outperformed others with accuracy, PRC and ROC score of 0.861, 0.781 and 0.946. Figure 4.9 visually compares the ROC and PRC of the three experiments.

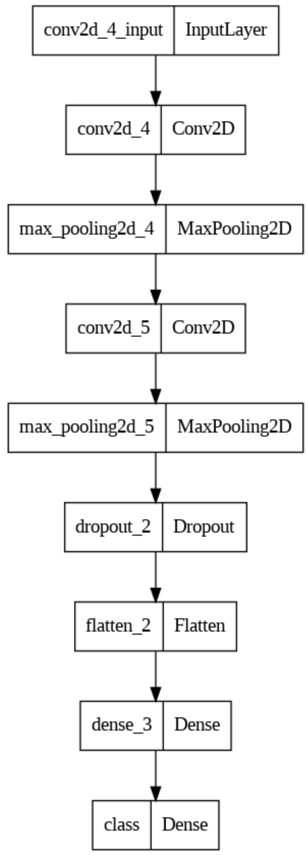

Figure 4. 7: CNN Model Structure

Table 4. 7: ICURP Performance on Dataset A

Window Size	**Accuracy**	**Precision**	**Recall**	**PRC**	**ROC**
24-h	0.795	0.759	0.863	0.839	0.862
48-h	**0.816**	**0.788**	**0.863**	**0.866**	**0.882**
<u>72-h</u>	<u>0.806</u>	<u>0.764</u>	<u>0.884</u>	<u>0.853</u>	<u>0.876</u>

Table 4. 8: ICURP Performance on Dataset B

Window Size	Accuracy	Precision	Recall	PRC	ROC
24-h	0.847	0.562	0.9375	0.731	0.935
48-h	0.861	0.588	0.947	0.781	0.946
72-h	**0.845**	**0.558**	**0.959**	**0.786**	**0.947**

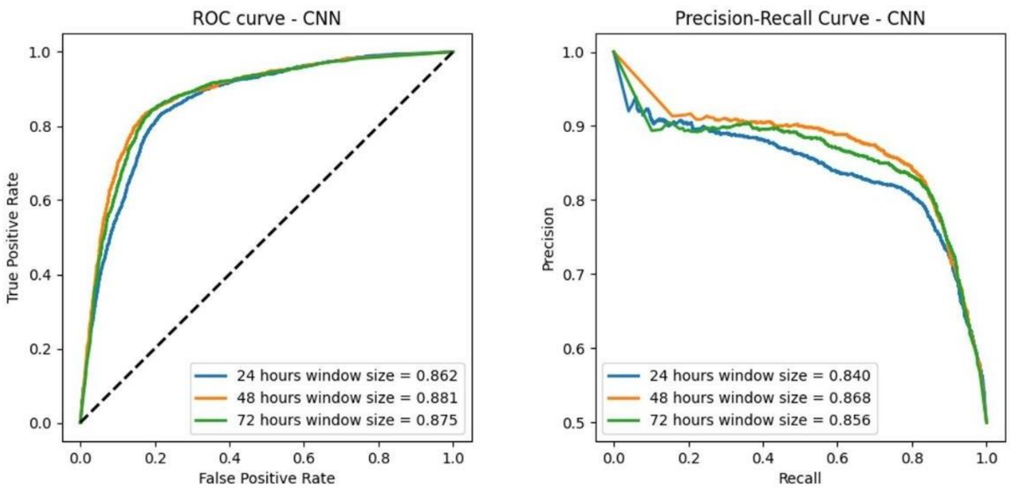

Figure 4. 8: ICURP's ROC & PR curves on dataset A

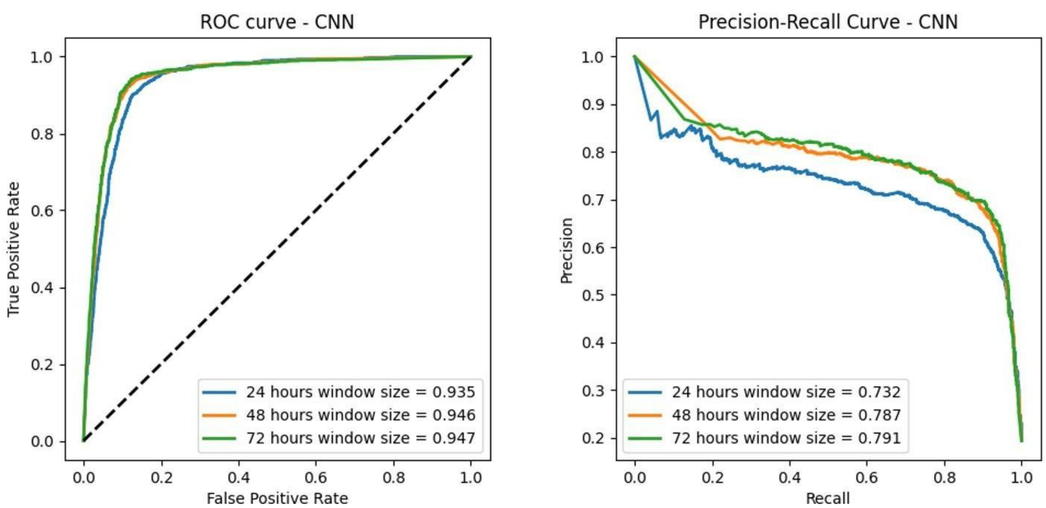

Figure 4. 9: ICURP's ROC & PR curves on dataset B

Figure 4.10 plots the training and validation process for the 48-hour window size. This plot shows that model starts to overfit on validation data after 50th epoch. Figure 4.11 shows the prediction results for few ICU stays samples using dataset A on last 48-hour ICU data. Only static features are shown here for our own convenience. Demographics features include Age, Gender, Ethnicity and Insurance while Diagnoses features include diagnoses code and diagnoses count. Period length represents the time spent by patient in particular ICU stay on hourly basis. Prediction label shows the output/result; 1 represents positive outcome (Needs Readmission), 0 represents negative outcome (Needs No Readmission).

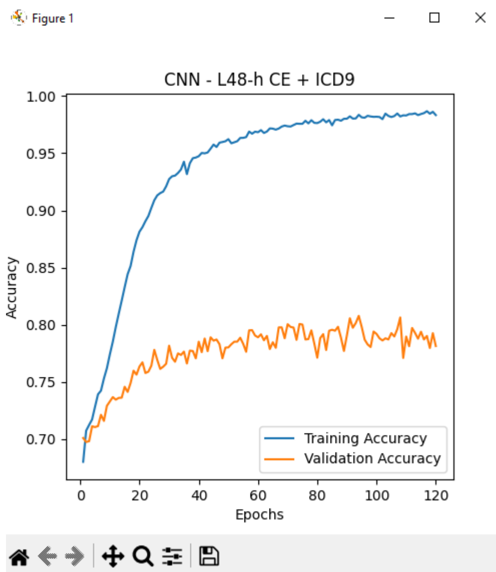

Figure 4. 10: ICURP – Model Training Plot

Period Length	Diagnoses	Diagnoses Count	Procedures	Procedures Count	Age	Gender	Ethnicity	Insurance	prediction _label
430.1	6143, 72886, 0389, 99592, 56781, 9982, 99749, 11289, 99800, 2851, 5849, 59589, 23989, 6202, V6441, 5680, E8700, E8782, 59971, 7850, 42731, 4019, 6238, 2720,	34	6849, 6561, 5459, 5421, 4561, 4591, 5781, 5412, 4591, 5412, 5459, 543, 543, 8663, 5732, 4319, 4524, 4525, 9672, 3893,	21	300.21	1	0	2	1
48.53	389, 78552, 5761, 5762, 1561, 5771, 5849, 99592, 42731, 4019, 2449, 25000, 35800	13	5110, 9705, 3893, 3891	4	83.27	1	4	2	1
16.84	29181, 30391, 3441, V600	4		0	45.88	2	4	4	1
332.5	48282, 51884, 99689, 27952, 5168, 73313, 2536, 79902, 33819, 33829, 78901, 7245, V1062, 4940, V462, 73300, 28529, V1251, V5861, V5865, 7905, 53081, 53085, 4019,	25	14, 3893	2	58.46	2	4	2	1
27.11	41401, 4111, 9971, 42731, 42789, 25080, 2720, 412	8	3612, 3615, 3961	3	72.13	2	0	2	0
22.66	40301, 42830, 5855, 4280, 25061, 3572, 78906	7	3995	1	44.32	2	3	2	1
131.5	9726, 5849, 78559, 4280, V422, 2767, 570, 42731, 78551	9	3893, 3893, 4523	3	86.41	1	4	2	0
264.3	03849, 51881, 4820, 5849, 9986, 5100, 99811, 2851, 2639, 1985, 99592, 4472	12	3421, 311, 7749, 3409, 9672, 9604, 9605, 3324, 3324, 4311, 966, 3893, 3891, 3323, 9915, 9904	16	65.92	1	4	2	1
26.63	41401, 4111, 496, 4019, 3051, 53081, 60000, V1051	8	3615, 3722, 3614, 8856, 8853, 8845, 3961, 8872	8	69.76	2	4	2	1
1055	20152, 99685, 79902, 51881, 486, 5990, 5849, 2762, 2761, 5859, 2869, 2449, 7245, 45821, 28522, 37034, 6110, 30000, 36281,	22	4105, 92, 3893, 9925, 9604, 9672, 966, 9915, 3895, 3995, 9971, 9604, 9672, 331, 331,	19	38.43	1	4	2	1

Figure 4. 11: Readmission Prediction Results using ICURP

Discussion

This section presents the detail discussion on results achieved from experiments which are described earlier in this chapter. It compares the evaluation results of experiments among themselves and also compares the achieved results with the state-of-the-art. Performance metrics used for comparison are accuracy, ROC, and PRC.

Comparison among Experimented Techniques

We have combined results of all experiments and have shown them in Table 4.9. Since last 48-hour ICU data has given better results for all experimented techniques, so we have reported only those results for the sake of comparison. All models compared here are evaluated using test dataset A.

From the experimental results of four conventional models (SVM, LR, NB & KNN), we can see that SVM is able to achieve high ROC and PRC score which is good, but since the dataset we are handling with, is time series data. There are new techniques available which can process temporality in data and we can achieve even more better results through them so, traditional ML techniques are not recommend here. LSTM and GRU have the ability to process temporal data sequentially, so they achieved better results than SVM. CNN model from ICURP's framework has achieved even more better results than RNN based models.

Table 4. 9: Comparison of all Experimented Techniques

Classifier	Accuracy	PRC	ROC
ICURP	**0.816**	**0.866**	**0.882**
LSTM	0.775	0.815	0.842
GRU	0.772	0.788	0.826
SVM	0.719	0.776	0.789
LR	0.702	0.765	0.774
KNN	0.687	0.741	0.756
NB	0.642	0.733	0.714

So, we can easily conclude that ICD-9 diagnosis embeddings and last 48-hour chart events features if used altogether, they are best at predicting patients' readmission before their discharge. Among the results of all techniques, ICURP achieved highest results for all the three chosen performance metrics accuracy, PRC and ROC. It has achieved great results with approximate 4% increase in accuracy, 5% increase in PRC and 4% increase in ROC scores than LSTM which has got the second highest scores.

Confusion matrixes are also created using LSTM and ICURP in Table 4.10 and Table 4.11 respectively. While prediction using ICURP, we see increase in true positive and true negative instances as compared to prediction using LSTM. And also, false negative and false positive instances have been reduced in great number contrary to LSTM, putting an impact on increase in overall performance. One more important aspect to be noticed is that, false negative instances have seen great reduction than false positive instances. This is our preferred metric, means that, patients with illness should be given higher chances to avail ICU facilities.

Table 4. 10: Confusion Matrix- LSTM

Actual	Predicted	
	0	**1**
0	2174	703
1	594	2282

Table 4. 11: Confusion Matrix- ICURP

Actual	Predicted	
	0	**1**
0	2210	667
1	394	2482

Comparison with the State-of-the-Art

The comparison of different methodologies from the past six years literature on MIMIC dataset is shown in Table 4.12. The experimental results reveal that the ICURP framework acquired a significantly better classification performance as compared to other studies that used time series data and deep learning models in their research. The experiment performed in [16] show better performance than our framework. Their experiment targets a specific disease while our ICURP framework does not target any specific disease, it is more generalized. The performance of RF model is also very high in [31] and it is equivalent to ICURP's performance but as per our understanding, they haven't used time series data. Moreover, model training was performed on a dataset that included records of patients from both institutions i.e., internal and external validation.

Table 4. 12: Comparison with the State of the Art on MIMIC Dataset

Year	Author	Diagnosis Specific	Technique	Accuracy (ACC %)	AUC (%)
2023	**Our**	**No**	**ICURP Framework**	**0.816**	**ROC=0.882 PRC=0.866**
2023	[44]	No	BERT		ROC=0.75 PRC=0.30
2023	[14]	Yes	RNN+LSTM		ROC=0.86 PRC=0.71
2022	[37]	No	SVM with the radial basis function kernel (RBF)		ROC=0.74
2022	[16]	Yes	Neural Network and using Decay Replay Mining	0.84	ROC=0.93
2022	[15]	Yes	RNN+LSTM		ROC=0.82 PRC=0.57
2021	[35]	No	Logistic Regression (LR)		ROC=0.76
2020	[43]	No	Fusion-LSTM		ROC=0.67 PRC= 0.79
2020	[42]	No	RNN+ neural ordinary diff. equations (ODE) with time-aware attention		ROC=0.74
2020	[33]	No	Random Forest	0.65	ROC=0.66
2019	[32]	No	NLP + Logistic Regression (LR)		ROC=0.75
2019	[40]	No	LSTM+CNN		ROC=0.79
2019	[31]	No	Random Forest with extended feature set	0.84 - 0.85	ROC=0.86 - 0.88
2018	[38]	No	Extreme Gradient Boosting (XGBoost)		ROC=0.71
2018	[28]	No	Extreme Gradient Boosting (XGBoost)		ROC=0.76
2018	[30]	No	Gradient Boosted Machine (GBM)		ROC=0.71 – 0.78

So, their results could have significant change. While, from all the remaining articles mentioned here, our proposed ICURP framework outperformed others with ACC=0.816, PRC=0.866 and ROC=0.882.

Proposed ICURP framework results can be validated by analyzing the characteristics of basic metrics shown in Table 4.13. Table compares the characteristics of four basic metrics TP, FN, FP and TN counts using their mean values. Patient ICU stay length has the maximum values for positively predicted cases means that patients who are predicted positive have the maximum ICU stay length. Similarly, the same trend is been observed for diagnoses count. Age and procedure count also have the maximum values for true positives. This validates our achieved results as patients diagnosed with multiple diseases have more probability of needing ICU readmission.

Table 4. 13: Characteristics of TP, FN, FP and TN (Mean Values)

Variables	TP	FN	FP	TN
Period Length	120.79	103.00	129.44	72.70
Age	85.33	78.93	77.14	69.40
Diagnoses Count	15.33	13.73	13.75	10.28
Procedure Count	6.41	5.79	5.30	3.84

From the above discussion, we can conclude that deep learning models perform better than conventional models with respect to accuracy, PRC and ROC. We can also conclude that our proposed ICURP framework has outperformed all other experimented models also acquired significantly better performance as compared to the state-of-the-art.

Chapter 5

CONCLUSION AND FUTURE WORK

The cost of intensive care is huge, which necessitates careful thought regarding when patients should be discharged or transferred to lower-level ward care. This decision is crucial to optimize the allocation of resources. However, discharging a patient from the ICU too early carries the potential risk of inadequate monitoring and care, often leading to readmission to the ICU. Therefore, it is important to strike a balance in determining the appropriate timing for step-down, ensuring that patients receive optimal care while minimizing the likelihood of readmission.

The application of ML techniques in ICUs has shown promising advancements in detecting high-risk events at an early stage. By leveraging ML techniques, clinicians can better analyze and interpret ICU data, enabling more timely and informed decision-making. This research is carried to solve the underlying problem of 30-day ICU readmission prediction. We have compared different conventional ML and deep learning models. We have analyzed that chart events data incorporated with ICD-9 diagnosis embeddings, are best at predicting readmission. Convolutional neural network architecture trained on MIMIC-III dataset has outperformed other models for prediction of 30-day ICU readmission.

By adopting this data-driven approach, we aspire to alleviate the premature discharge or transfer of patients. Correctly identified patients can be benefitted with long stays in the ICU, allowing them to receive the necessary medical attention. On the other hand, by provision of timely decision-making support to clinicians, proposed ICURP framework will contribute to reducing unnecessary readmissions, thereby minimizing the financial burden imposed on healthcare facilities. The implementation of our approach can optimize resource allocation, improve patient outcomes, and enhance the overall efficiency of ICU care.

Our next objective is to investigate alternative methods for effectively combining all data modalities to improve the performance of prediction results. We also aim to explore other relevant data features that can be helpful to improve our results as MIMIC-III dataset is huge and we have just used a little data from it.

REFERENCES

1. Marshall, J.C., et al., *What is an intensive care unit? A report of the task force of the World Federation of Societies of Intensive and Critical Care Medicine.* Journal of critical care, 2017. **37**: p. 270-276.
2. Loreto, M., T. Lisboa, and V.P. Moreira, *Early prediction of ICU readmissions using classification algorithms.* Computers in biology and medicine, 2020. **118**: p. 103636.
3. Kramer, A.A., T.L. Higgins, and J.E. Zimmerman, *The association between ICU readmission rate and patient outcomes.* Critical care medicine, 2013. **41**(1): p. 24-33.
4. Ponzoni, C.R., et al., *Readmission to the intensive care unit: incidence, risk factors, resource use, and outcomes. A retrospective cohort study.* Annals of the American Thoracic Society, 2017. **14**(8): p. 1312-1319.
5. Desautels, T., et al., *Prediction of early unplanned intensive care unit readmission in a UK tertiary care hospital: a cross-sectional machine learning approach.* BMJ open, 2017. **7**(9): p. e017199.
6. Chen, L.M., et al., *Patients readmitted to the intensive care unit during the same hospitalization: clinical features and outcomes.* Critical care medicine, 1998. **26**(11): p. 1834-1841.
7. Singer, D.E., et al., *Unexpected readmissions to the coronary-care unit during recovery from acute myocardial infarction.* New England Journal of Medicine, 1981. **304**(11): p. 625-629.
8. McIlvennan, C.K., Z.J. Eapen, and L.A. Allen, *Hospital readmissions reduction program.* Circulation, 2015. **131**(20): p. 1796-1803.
9. FY2018-IPPS-Final-Rule-Data-Files. 2017; Available from: https://www.cms.gov/medicare/medicare-fee-for-service-payment/acuteinpatientpps/acute-inpatient-files-for-download-items/fy2018-final-rule-correction-notice-files.
10. Wang, H.-j., et al., *Preventable readmission to intensive care unit in critically ill cancer patients.* World Journal of Emergency Medicine, 2018. **9**(3): p. 211.
11. Baillie, C.A., et al., *The readmission risk flag: using the electronic health record to automatically identify patients at risk for 30-day readmission.* Journal of hospital medicine, 2013. **8**(12): p. 689-695.
12. Kansagara, D., et al., *Risk prediction models for hospital readmission: a systematic review.* Jama, 2011. **306**(15): p. 1688-1698.
13. Shams, I., S. Ajorlou, and K. Yang, *A predictive analytics approach to reducing 30-day avoidable readmissions among patients with heart failure, acute myocardial infarction, pneumonia, or COPD.* Health care management science, 2015. **18**: p. 19-34.
14. Kessler, S., et al., *Predicting readmission to the cardiovascular intensive care unit using recurrent neural networks.* Digital Health, 2023. **9**: p. 20552076221149529.
15. Moazemi, S., et al., *Evaluating a recurrent neural network model for predicting readmission to cardiovascular ICUs based on clinical time series data.* Engineering Proceedings, 2022. **18**(1): p. 1.

16. Pishgar, M., et al., *Prediction of unplanned 30-day readmission for ICU patients with heart failure.* BMC Medical Informatics and Decision Making, 2022. **22**(1): p. 1-12.
17. Kim, H., et al., *Scheduled and unscheduled hospital readmissions among diabetes patients.* The American journal of managed care, 2010. **16**(10): p. 760.
18. Nijhawan, A.E., et al., *Half of 30-day hospital readmissions among HIV-infected patients are potentially preventable.* AIDS patient care and STDs, 2015. **29**(9): p. 465-473.
19. McAdams-DeMarco, M.A., et al., *Frailty and early hospital readmission after kidney transplantation.* American journal of transplantation, 2013. **13**(8): p. 2091-2095.
20. Curto, S., et al. *Predicting ICU readmissions based on bedside medical text notes*. in *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. 2016: IEEE.
21. Harutyunyan, H., et al., *Multitask learning and benchmarking with clinical time series data.* Scientific data, 2019. **6**(1): p. 96.
22. Choi, Y., C.Y.-I. Chiu, and D. Sontag, *Learning low-dimensional representations of medical concepts.* AMIA Summits on Translational Science Proceedings, 2016. **2016**: p. 41.
23. Johnson, A.E., et al., *MIMIC-III, a freely accessible critical care database.* Scientific data, 2016. **3**(1): p. 1-9.
24. Syed, M., et al. *Application of machine learning in intensive care unit (ICU) settings using MIMIC dataset: systematic review.* in *Informatics*. 2021: MDPI.
25. Bennett, D. and J. Bion, *ABC of intensive care. Organisation of intensive care.* BMJ (Clinical research ed.), 1999. **318**: p. 1468-70.
26. Messaoudi, Z., *The 3 main steps of Machine Learning.* 2020.
27. Pandian, S., *Time Series Analysis and Forecasting | Data-Driven Insights.* 2023.
28. Pakbin, A., et al. *Prediction of ICU readmissions using data at patient discharge*. in *2018 40th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*. 2018: IEEE.
29. Min, X., B. Yu, and F. Wang, *Predictive modeling of the hospital readmission risk from patients' claims data using machine learning: a case study on COPD.* Scientific reports, 2019. **9**(1): p. 2362.
30. Rojas, J.C., et al., *Predicting intensive care unit readmission with machine learning using electronic health record data.* Annals of the American Thoracic Society, 2018. **15**(7): p. 846-853.
31. McWilliams, C.J., et al., *Towards a decision support tool for intensive care discharge: machine learning algorithm development using electronic healthcare data from MIMIC-III and Bristol, UK.* BMJ open, 2019. **9**(3): p. e025925.
32. Li, Z., et al., *Early prediction of 30-day ICU re-admissions using natural language processing and machine learning.* arXiv preprint arXiv:1910.02545, 2019.
33. Assaf, R. and R. Jayousi. *30-day hospital readmission prediction using MIMIC data*. in *2020 IEEE 14th International Conference on Application of Information and Communication Technologies (AICT)*. 2020: IEEE.

34. Liu, W., et al., *Predicting 30-day hospital readmissions using artificial neural networks with medical code embedding.* PloS one, 2020. **15**(4): p. e0221606.
35. Moerschbacher, A. and Z. He, *Building prediction models for 30-day readmissions among icu patients using both structured and unstructured data in electronic health records.* medRxiv, 2021: p. 2021.08. 10.21261858.
36. Shi, K., et al. *Predicting unplanned 7-day intensive care unit readmissions with machine learning models for improved discharge risk assessment*. in *AMIA Annual Symposium Proceedings*. 2022: American Medical Informatics Association.
37. Orangi-Fard, N., A. Akhbardeh, and H. Sagreiya. *Predictive model for icu readmission based on discharge summaries using machine learning and natural language processing*. in *Informatics*. 2022: MDPI.
38. Rafi, P., A. Pakbin, and S.K. Pentyala, *Interpretable deep learning framework for predicting all-cause 30-day ICU readmissions.* Texas A&M University, 2018.
39. Wang, H., et al., *Predicting hospital readmission via cost-sensitive deep learning.* IEEE/ACM transactions on computational biology and bioinformatics, 2018. **15**(6): p. 1968-1978.
40. Lin, Y.-W., et al., *Analysis and prediction of unplanned intensive care unit readmission using recurrent neural networks with long short-term memory.* PloS one, 2019. **14**(7): p. e0218942.
41. Ashfaq, A., et al., *Readmission prediction using deep learning on electronic health records.* Journal of biomedical informatics, 2019. **97**: p. 103256.
42. Barbieri, S., et al., *Benchmarking deep learning architectures for predicting readmission to the ICU and describing patients-at-risk.* Scientific reports, 2020. **10**(1): p. 1111.
43. Zhang, D., et al., *Combining structured and unstructured data for predictive models: a deep learning approach.* BMC Medical Informatics and Decision Making, 2020. **20**(1): p. 1-11.
44. Sheetrit, E., M. Brief, and O. Elisha, *Predicting Unplanned Readmissions in the Intensive Care Unit: A Multimodality Evaluation.* arXiv preprint arXiv:2305.08139, 2023.
45. Johnson, A.E.P., T.; Mark, R., *MIMIC-III Clinical Database (version 1.4). PhysioNet.* 2016.
46. Brown, S.E., S.J. Ratcliffe, and S.D. Halpern, *An empirical derivation of the optimal time interval for defining ICU readmissions.* Medical care, 2013. **51**(8): p. 706.
47. O'Shea, K. and R. Nash, *An introduction to convolutional neural networks.* arXiv preprint arXiv:1511.08458, 2015.
48. Park, J., D. Yi, and S. Ji, *Analysis of recurrent neural network and predictions.* Symmetry, 2020. **12**(4): p. 615.
49. Le, X.-H., et al., *Application of long short-term memory (LSTM) neural network for flood forecasting.* Water, 2019. **11**(7): p. 1387.
50. ArunKumar, K., et al., *Comparative analysis of Gated Recurrent Units (GRU), long Short-Term memory (LSTM) cells, autoregressive Integrated moving average (ARIMA), seasonal autoregressive Integrated moving average (SARIMA)*

for forecasting COVID-19 trends. Alexandria engineering journal, 2022. **61**(10): p. 7585-7603.