# Predictive Modeling for Hospital Readmission Risk Using XGBoost and LightGBM Techniques with Class Imbalance

**Maragadhavalli Meenakshi M, Jayapratha N, Vijaysurya M, Lingeshwaran G, Prethega SN, Hemachandran S, Priyadarshani A and Abinaya S**

Sri Manakula Vinayagar Engineering College, vijaysuryapdy@gmail.com

## Abstract

This paper reports an end-to-end machine-learning framework to predict the likelihood of patient readmission. Scaling the process and proper data management required scalability for SQLAlchemy and PostgreSQL. Additionally, an ETL pipeline was created to integrate and transform structured data from EHRs. This, then, gathered all the essential knowledge related to a patient, while model development included training followed by the comparison of many classifiers: specifically, Random Forest, XGBoost, and LightGBM. The LightGBM algorithm outperformed the others, with an AUC-ROC of 0.88 and an accuracy of 83%, both of which were rather good. As a result, SHAP (SHapley Additive exPlanations) values made the model interpretable, making the predictions accessible to clinicians as well. To further promote data-driven clinical decision-making, a Streamlit web application was made available that allowed real-time interaction with the model itself.

**Keywords:** Patient Readmission, SQLAlchemy, PostgreSQL, ETL pipeline, Random Forest, XGBoost, LightGBM, healthcare analytics

## 1. Introduction

Hospital readmission is a 30-day factor that has gained much importance for measuring healthcare delivery quality and efficiency in using resources. Identifying the patients most likely to be readmitted helps the hospitals optimize resource utilization. This data-driven foundation for real clinical decision-making has been made possible by machine learning's strong predictive analytics capabilities in the healthcare industry. It all, however, rides behind careful data engineering, interpretability, and most importantly, user-accessibility of the interface to release of ML models into real-world settings.

This research used an approach that integrates SQL-based data handling, feature engineering, model optimization, and real-time deployment into a full solution for readmission risk prediction. Utilizing SQLAlchemy and PostgreSQL, we designed an ETL pipeline that processes, structures, and stores large-scale EHR data. Our model showed robust predictive accuracy, further supported by an interpretable interface, allowing clinicians to understand key risk factors.

## 2. Literature review

To build a solid foundation for the literature on hospital-based readmission prediction, this feature has been found with numerous considerations, including data processing,

class balance, model selection, interpretability, and even healthcare deployment.

### 2.1. Data processing and feature engineering:

Since the format of the data obtained from EHR systems varies greatly, it is imperative that it be processed systematically; this was previously demonstrated in experiments where feature engineering was successful. Numerous predictor variables will be taken into consideration, including the patient's medical history, lab results, and demographic information. More encoding techniques such as one-hot encoding, categorical aggregation, and even ICD codes will also arise to capture complexity in diagnosis; index scales of comorbidities will also have some quantified scales.

Schuller, B., et al. (2004) [3] Effective feature engineering has proven to be efficient in enhancing prediction accuracy in healthcare models, especially by taking in varied patient data such as demographics and medical history.

### 2.2 Handling Class Imbalance:

Gong, H., et al. (2023) [7] Class imbalance problems are critical in healthcare, but positive cases are often underrepresented, and in this particular scenario, SMOTE performs exceptionally well in synthetic sample generation.

This domain includes the imbalance that exists in healthcare with readmission cases versus those not readmitted. The class imbalance discussed here may result in biased models that underestimate rare events. SMOTE balances a dataset with the synthetic minority oversampling technique by making synthetic samples of the minority class. Since cross-validation is recommended and more specifically 5-fold, overfitting must be avoided.

### 2.3. Model Selection:

Tripathi, S., et al. (2017) [9] Ensemble methods, particularly Random Forest and LightGBM, are very popular in healthcare analytics for their ability to deal with high-dimensional data and improve predictive performance.

Structured data is quite robust and scalable, so the ensemble method of Random Forest, XGBoost, and LightGBM suits them well. Any model that works efficiently with missing data, with categorical variables, and those that use nonlinear relationships mostly go well with high dimensional healthcare data.

### 2.4. Model Interpretability:

It requires interpretability in the prediction done by the model for its clinical deployment. The SHAP (SHapley Additive exPlanations) values are very helpful for feature importance such that the clinicians can have an idea of what causes the readmission risk more often. This sort of interpretability is extremely crucial for supporting clinical decision-making and gaining trust within the ML systems.

Picard, R.W. (2000) [5] Machine learning models' interpretability plays a critical role in clinical applications so that, by understanding the rationale of the predictions the healthcare providers may gain even more trust in these systems.

### 2.5. Real-time Deployment:

For the predictive models of recent research to use web frameworks like Streamlit towards achievable user interfaces to enable

healthcare professionals to interact with an ML model in real-time, there must be effective user interfaces available for real-time deployment within the hospitals.

Matsumoto, D., & Hwang, H.C. (2013) [10] User-friendly interfaces that enable clinicians to quickly enter data and view model predictions in real-time should be the main focus of predictive model application in clinical contexts. An interface that makes data entry and retrieval easier makes it possible to apply ML insights in healthcare operations.

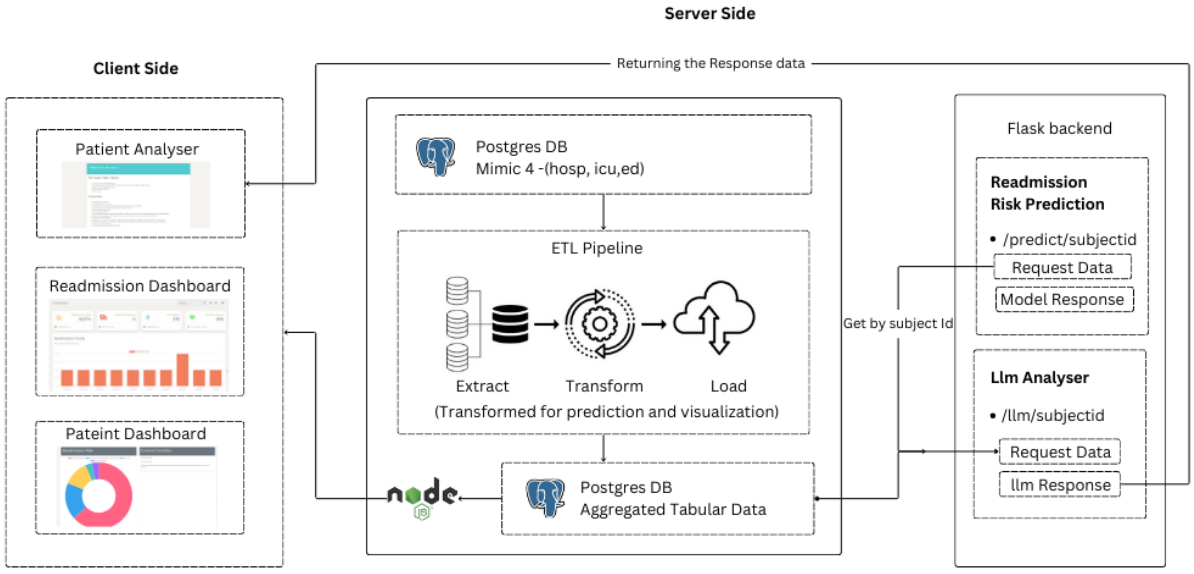


**Figure 1.** ETL Processing Workflow Architecture

## 3. Proposed System

### 3.1 Data collection and storage

The corresponding demographics, ICD-9/10 diagnoses, lab data, and admission history are retrieved from the hospital's EHR system. With the aid of SQLAlchemy, PostgreSQL was utilized effectively for both structured data retrieval and storage. It was handled by an ETL pipeline that was completed in Pandas, where the cleaning process, as well as all transformation steps, replaced the missing data normalized.

### 3.2 Feature Engineering

We created features to be sure that models were predictive:

**ICD Code Processing:** The ICD codes were encoded and bucketed, which made sure that the dimensions of the features reduce appropriately without losing any form of diagnostic understanding. Categorical features worked well along with tree-based models along with one-hot encoding.

**The Charlson Comorbidity Index (CCI):** It is a metric that is used to estimate the severity of a patient's comorbidities. This has been a high-risk readmission predictor. Clinical research has also made extensive use of this index to evaluate the likelihood of unfavourable outcomes.

The major demographics included in the analysis were age and gender along with the lab values which included sodium and potassium levels as that had a lot of information on the state of health and risk for patients.

It encompasses feature extraction process, where ICD encoding, calculations for the Charlson Comorbidity Index, and lab result analysis are placed into predictive variables. These make the core constituents of our model in [Figure 1] on how data is transformed stepwise.

### 3.3 Addressing Class Imbalance

To avoid class imbalance, SMOTE was applied such that new samples were generated by the minority class to represent both sides and balance them. In addition, a 5-fold cross-validation technique was used, which increases generalization and reduces overfitting.

### 3.4 Model Selection and Development

This section compares the performance of the classifiers Random Forest, XGBoost, and LightGBM in predicting readmission risk. As shown in [Figures 2 and 3], model evaluation metrics for XGBoost and LightGBM are provided. The figures will help us visualize how well the classifiers are performing, including their accuracy, AUC-ROC, and interpretability, where LightGBM is found to be the best model to be deployed.

### 3.5 Hyperparameter Optimization

Hyperparameters were tuned by RandomizedSearchCV with K-fold cross-validation.The important parameters were:

1. **Random Forest:** n_estimators, max _depth, and max_features needed tuning to find the optimal balance between complexity and performance.
2. **XGBoost:** Class weight scaling subsample, max_depth, and eta were tuned.
3. **LightGBM:** learning_rate, min data in leaf, and num_leaves were tuned to be as computationally efficient as possible without losing accuracy.

### 3.6 Model Interpretation and Feature Importance

The study used SHAP values to explain the importance of features and how factors like the Charlson Comorbidity Index or lab results influenced the predictions. Since SHAP provides transparency, model findings can be presented in a way that makes sense to users, particularly clinicians.

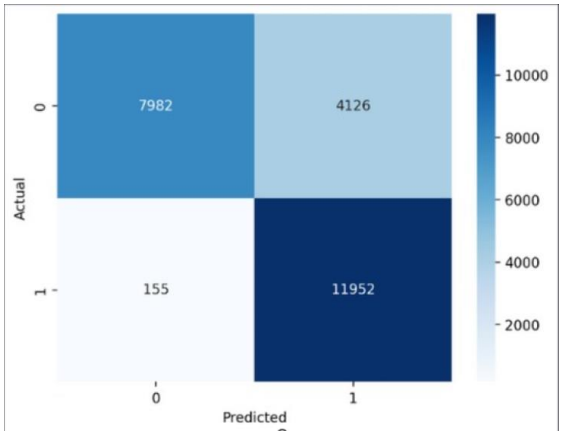


**Figure 2.** Model Evaluation of XgBoost

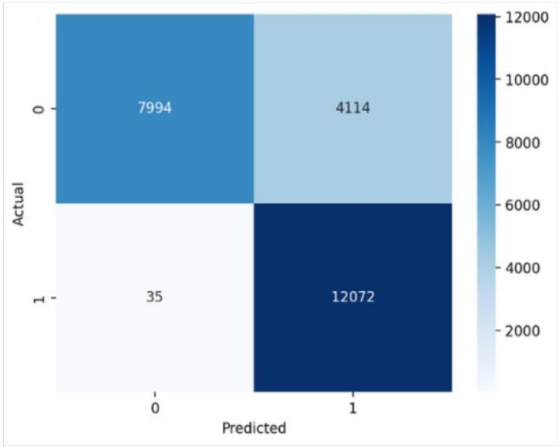


**Figure 3.** Model Evaluation of LightGBM

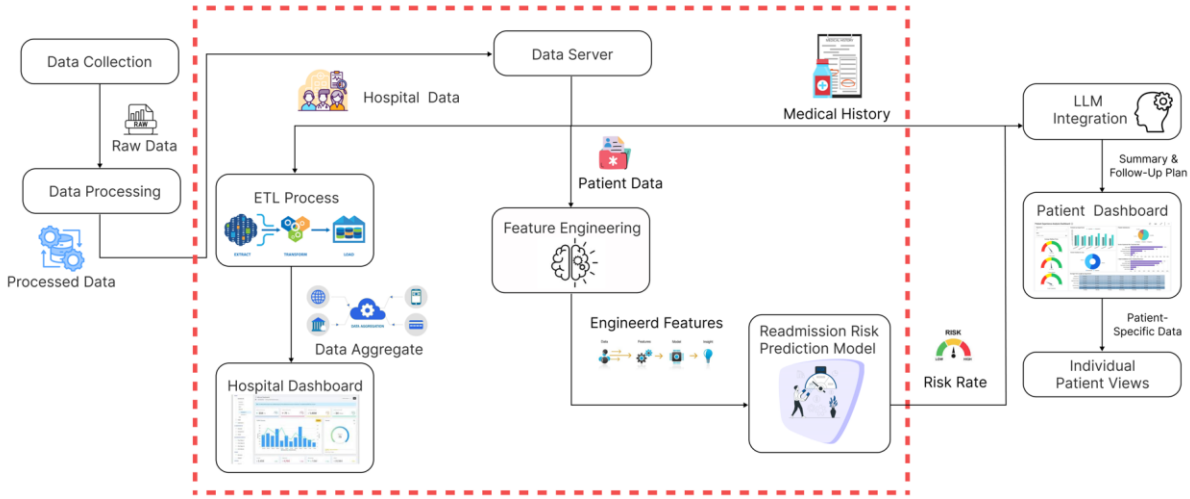


**Figure 4.** Overall Architecture of **Hospital Readmission Risk Prediction**

## 4. Architecture Diagram

[Figure 4] This diagram shows all the architecture, from data gathering to the prediction model. It captures all the procedures in the ETL procedure and data structuring and integration of real-time deployment. It focuses its flow from the storage of data in PostgreSQL onto interaction with the web interface, supporting clinical decision-making.

## 5. Predictive Model Deployment

The architecture of the system for the real-time deployment of predictive models in the clinical environment that includes the integration with the web interface. This process allows clinicians to input the patient data and allow for immediate readmission to facilitate smooth clinical workflows.

## 6. Result

The results from the model evaluation [Figure 2, Figure 3] show that the AUC-ROC of LightGBM has been observed to be highest at 0.88 and the accuracy 83%. Real-time deployment demonstrated the capability of the model to make strong predictions using unseen data samples, with sound outcomes for clinical purposes.

## 7. Challenges

Data Integration: Incompatibility of SQL, Pandas, and Scikit-learn led to the design of ETL being relatively challenging for easy data flow.

1. **Real-time Processing:** It should provide accurate output in real-time, thus requiring optimization of both the ETL pipeline and deployment architecture.
2. **User Interface:** An intuitive interface in Streamlit for non-technical users was important to support clinical use.
3. **Data Security:** This was achieved by encryption protocols concerning confidentiality and integrity since the data involves highly sensitive patients.

## 8. Web Interface

The web interface is user friendly, therefore, allowing clinicians to interact with the model directly by feeding it patient data; this directly supports practical and efficient real-time readmission risk predictions in clinical settings.

## 9. Conclusion and Future Work

This paper suggests a data-driven readmission risk predictor that makes use of strong feature engineering coupled with the management mechanism of SQL data and a real-time-deploying mechanism. In the future, this system can be extended by including more feature sets according to extra patient history metrics.

Salovey, P.E., & Sluyter, D.J. (1997) [8] Future research in healthcare predictive modeling should focus on improving model robustness and exploring more feature sets that can provide more insights into patient risk factors.

Future work will focus on optimizing large datasets for model optimization and better usability with real-time interfaces.

## References


[1] Waibel, A., & Fugen, C. (2008). *Spoken language translation*. IEEE Signal Processing Magazine.

[2] Jia, Y., et al. (2019). *Direct speech-to-speech translation with a sequence-to-sequence model*. ArXiv preprint.

[3] Schuller, B., et al. (2004). *Speech emotion recognition combining acoustic features*. IEEE International Conference.

[4] Arik, S., et al. (2018). *Neural voice cloning with a few samples*. Advances in Neural Information Processing.

[5] Picard, R.W. (2000). *Affective computing*. MIT Press.

[6] Dignum, V. (2018). *Ethics in artificial intelligence*. Ethics and Information Technology.

[7] Gong, H., et al. (2023). *Multilingual speech-to-speech translation into multiple target languages*. ArXiv preprint.

[8] Salovey, P.E., & Sluyter, D.J. (1997). *Emotional development and emotional intelligence*. Basic Books.

[9] Tripathi, S., et al. (2017). *Using deep and convolutional neural networks*. AAAI Conference.

[10] Matsumoto, D., & Hwang, H.C. (2013). *Cultural similarities and differences in emblematic gestures*. Journal of Nonverbal Behavior.

[11] Devlin, J. (2018). *BERT: Pre-training of deep bidirectional transformers*. ArXiv preprint.

[12] Kanimozhi, P., et al. (2024). *Revolutionizing Hearing Health: Mobile-based Audiometry*. IEEE.

[13] Conneau, A. (2019). *Unsupervised cross-lingual representation learning*. ArXiv preprint.

[14] Granroth-Wilding, M., & Toivonen, H. (2019). *Unsupervised learning of cross-lingual symbol embeddings*. Proceedings of SCiL.

[15] Ephrat, A., et al. (2018). *Looking to listen at the cocktail party*. ArXiv preprint.