Predicting 30-Day Hospital Readmissions

Aryan Patil
Cornell Tech
Health Tech
ap2365@cornell.edu

Neelraj Patil
Cornell Tech
Health Tech
njp75@cornell.edu

Omkar Garad
Cornell Tech
Health Tech
omg22@cornell.edu

Abstract

Hospital readmissions within 30 days after discharge present a significant burden on healthcare systems, both financially and in terms of patient well-being. Reducing these readmissions is crucial for improving patient outcomes and reducing costs. In this project, we propose a machine learning–based approach to predict which patients are at risk of being readmitted within 30 days. Specifically, we will implement two models from scratch, i.e., Logistic Regression and Random Forest, and train them on a publicly available hospital readmission dataset (e.g., from Kaggle or the UCI Machine Learning Repository). Our approach builds on previous research that uses machine learning to enhance clinical decision-making; however, we will implement these algorithms fully from scratch and integrate them into a user-friendly web application.

To enable real-time risk assessment, we will design a Web app using Streamlit for the front end, allowing hospital administrators or clinical staff to enter relevant patient data. The back end includes our machine learning pipeline, where the final trained model (or both models for comparison) generates a probability score for readmission. We will evaluate our models using accuracy, F1 score, and ROC-AUC, with cross-validation and hyperparameter tuning to avoid overfitting or underfitting. We expect Random Forest to yield higher predictive performance, while Logistic Regression may prove easier to interpret, which is an essential consideration in healthcare settings. Broadly, our work aims to demonstrate how lightweight, interpretable machine learning solutions can be rapidly deployed to improve patient care and resource allocation, thereby contributing a practical solution to the healthcare community.

1. Introduction

Hospital readmissions within 30 days after discharge are a pressing concern for both healthcare providers and patients. Such readmissions often signify gaps in post-discharge care and can lead to higher financial costs for institutions, not to mention undue stress and potential health complications for the patient. By predicting which patients are at high risk of readmission, hospitals can allocate care more efficiently e.g., by ensuring close follow-up, scheduling timely appointments, or providing additional patient education. This project aims to develop a web-based application to facilitate this prediction, thus improving patient outcomes and lowering overall healthcare costs.

The primary machine learning methods we plan to use are Logistic Regression and Random Forest, both implemented from scratch. The first model is logistic regression which provides an interpretable baseline model that is frequently used in clinical contexts. Its coefficients can shed light on how particular features (e.g., the number of past admissions, specific comorbidities) influence readmission risk. Next Random Forest is an ensemble approach that typically outperforms simpler models when it comes to capturing non-linear feature interactions. This can lead to higher predictive accuracy and more nuanced insights into complex clinical data. What is unique about our approach is the end-to-end pipeline, from raw clinical data to an interactive front-end tool. Additionally, instead of relying on high-level machine learning libraries (e.g., scikit-learn) for the core algorithms, we will be implementing both Logistic Regression and Random Forest from scratch, giving us finer control over the training process and a deeper understanding of model mechanics.

Predictive modeling for hospital readmissions has been explored by healthcare organizations and researchers alike. Existing efforts often involve regression-based models (e.g., logistic regression, linear regression) with clinically relevant features or gradient boosting or ensemble methods that achieve high accuracy but sometimes lack interpretability. These studies have demonstrated moderate to strong performance in identifying high-risk patients. However, many of these approaches rely on pre-built machine learning libraries and focus solely on reporting predictive metrics rather than offering a user-friendly, real-time solution. Our project combines the theoretical rigor of from-scratch implementation with an emphasis on practical deployment via

a Streamlit app, bridging a gap between research prototypes and clinic-ready tools.

Our proposed pipeline consists of four main stages:

1. **Data Exploration:** We begin by collecting publicly available hospital readmission data. We will explore missing values, feature distributions, and correlations among variables (e.g., age, length of stay, discharge diagnosis) that might predict readmission risk.

2. **Preprocessing:** Steps include handling missing data, encoding categorical features (such as discharge status), and normalizing numerical features if necessary. This ensures the data is consistent and ready for model training.

3. **Model Training and Evaluation:** We will train both Logistic Regression and Random Forest from scratch on a training set, tuning hyperparameters (e.g., learning rate, number of trees, and tree depth). We plan to use Accuracy, F1-Score, and ROC-AUC to evaluate model performance. Cross-validation will help prevent overfitting and provide reliable comparisons.

4. **Deployment:** The final model (likely the one with the best trade-off between accuracy and interpretability) will be integrated into a Streamlit application. Hospital staff can enter relevant patient data into an online form and instantly receive a prediction score, thereby enabling more targeted post-discharge interventions.

Early identification of high-risk patients can substantially reduce readmission rates, benefiting both patients—by improving quality of care—and hospitals—by reducing financial penalties and optimizing resources. Nonetheless, there are ethical considerations related to patient data privacy and model bias. For instance, if the training data contain demographic biases, the model may inadvertently discriminate against certain patient groups. Ensuring compliance with health data privacy laws (such as HIPAA in the U.S.) is also paramount. Our project will address these concerns through careful feature selection, secure handling of any sensitive information, and transparent reporting of model performance across diverse patient subgroups.

2. Background

Hospital readmissions within 30 days of discharge represent a major challenge for healthcare systems worldwide, contributing to increased costs, patient morbidity, and preventable strain on resources [4]. Accurately identifying patients at high risk of early readmission enables targeted interventions—such as timely follow-up appointments or enhanced care coordination—that can both improve patient outcomes and reduce financial penalties imposed by payers [3]. The primary aim of this work is to develop an end-to-end machine learning pipeline, implemented from first principles, that not only achieves strong predictive performance but also supports real-time deployment through a lightweight web interface. By focusing on streamlined, interpretable models and transparent evaluation, we seek to bridge the gap between academic prototypes and clinical decision-support tools.

Early research on readmission prediction predominantly employed logistic regression, leveraging clinically selected covariates such as comorbidities and prior hospitalization history [5]. Over time, studies introduced more sophisticated ensemble and boosting approaches—such as gradient boosting machines and Random Forests—to capture nonlinear interactions among risk factors, often reporting modest gains in discrimination metrics like ROC-AUC [7]. More recently, deep learning models trained on high-dimensional electronic health record data have shown promise in modeling complex temporal patterns, though at the expense of interpretability and computational complexity [6]. These chronological advances highlight both the potential for improved accuracy and the trade-offs inherent in increasingly complex algorithms.

Despite consensus on the importance of accurate readmission risk estimation, there remains active debate regarding model interpretability versus predictive power. Healthcare stakeholders—including clinicians, administrators, and regulators—demand transparent models whose decision logic can be audited and explained, particularly when patient care decisions depend on algorithmic outputs [2]. Moreover, discrepancies in performance across demographic subgroups have raised concerns about potential bias and equity, underscoring the necessity for comprehensive reporting of model behavior on diverse patient populations.

While many prior works focus on leveraging high-level libraries for model development and concentrate on offline evaluation metrics, few address the practical challenges of implementing algorithms from scratch or deploying them in a user-centric application. Our proposed approach fills this gap by providing fine-grained control over algorithmic implementation—enhancing reproducibility and pedagogical insight—and by integrating both Logistic Regression and Random Forest into a Streamlit front end for instantaneous risk assessment. We anticipate that this combination of transparency, ease of deployment, and competitive performance will compare favorably to existing methods, offering a reusable framework for future clinical machine learning applications [1].

3. End-to-End ML Pipeline

3.1. Back-End

The Back-End is where we orchestrate all the essential machine learning tasks, from dataset handling to model deployment. By separating these functions from the user interface, we can maintain clean design principles while ensuring scalability and maintainability.

3.1.1 Data Collection, Exploration Processing

We plan to use a publicly available hospital readmission dataset, sourced from Kaggle or the UCI Machine Learning Repository. Below are the key points of our data plan:

The dataset will include numeric features (e.g., age, length of stay, number of prior admissions) and categorical features (e.g., discharge disposition, insurance type). Typically, these datasets contain tens of thousands of patient records released between 2014 and 2016 (specific dataset choice to be confirmed). We will analyze patient attributes to predict whether a patient will be readmitted within 30 days. In these datasets the ground truth readmission status (yes/no within 30 days) is already provided.

While conducting our data exploration we will use histograms, boxplots, bar charts, and correlation heatmaps as visualizations to help us identify key features influencing readmission. For feature analysis, we will investigate how variables such as comorbidities, number of prior admissions, or discharge codes correlate with 30-day readmissions.

We plan on utilizing several data preprocessing techniques to prepare our dataset effectively for modeling. First, missing data will be addressed through median imputation or, when necessary, by removing records exhibiting excessive missingness. Next, categorical variables such as "discharge status" will undergo encoding using either one-hot encoding or integer encoding, depending on their characteristics and cardinality. Additionally, we intend to apply normalization or scaling to continuous variables whose value ranges significantly differ, ensuring balanced feature contributions during training. Lastly, outlier detection and removal will be performed for features like lengths of stay or lab results, as extreme values in these variables may negatively impact model performance by distorting the training process.

3.1.2 Methods and Model Training

We propose employing two primary machine learning methodologies, namely Logistic Regression and Random Forest, each developed from scratch to address the predictive task at hand. Logistic Regression is particularly suitable for this study due to its widespread use in clinical research and its inherent interpretability through regression coefficients. This approach allows clinicians to readily assess how individual features influence the odds of patient readmission. To train and validate our logistic regression model, we will divide the dataset into training (80%) and testing (20%) subsets. If the dataset demonstrates significant class imbalance (i.e., a disproportionately larger number of "no readmission" cases relative to "yes" cases), sampling techniques or class weighting strategies will be employed to mitigate bias. Optimization will be performed using gradient descent to minimize cross-entropy loss.

In parallel, we will develop a Random Forest model, an ensemble method known for its ability to handle complex, non-linear relationships among features, often outperforming linear models in accuracy. Furthermore, Random Forest models naturally yield feature importance metrics, enhancing interpretability within healthcare contexts. Training will involve constructing individual decision trees, each based on bootstrapped samples of the dataset, while randomly selecting subsets of features at each node split to ensure diversity and robustness. Critical hyperparameters, such as the number of trees and maximum tree depth, will be optimized through cross-validation to prevent both overfitting and underfitting.

Both models will accept patient characteristics including numeric and categorical variables as inputs, and will output predictions either as probabilities of 30-day readmission (facilitating threshold-based classification) or as direct binary classifications ("readmitted" versus "not readmitted").

3.1.3 Model Evaluation

We will conduct a comprehensive set of experiments and evaluations to rigorously assess our proposed methods. For evaluating performance, we will utilize multiple metrics, including accuracy, as it provides an intuitive measure of correct predictions relative to total cases. Given the likelihood of class imbalance in the dataset, we will also prioritize the F1-score, which effectively balances precision and recall. Additionally, ROC-AUC will be employed to evaluate the models' abilities to distinguish between positive (readmitted) and negative (not readmitted) classes across various probability thresholds, further enriching our understanding of model performance.

Our experimental methodology includes a standard 80/20 train-test data split, supplemented by 5-fold cross-validation to perform hyperparameter tuning. Specifically, hyperparameters such as learning rate in Logistic Regression and number of trees or tree depth in Random Forest will be optimized through cross-validation to enhance generalization performance. To gain deeper insights into clinical applicability, we will also conduct detailed error analyses by examining false positives—patients flagged as high risk but not actually readmitted—and false neg-

atives—patients who were not flagged but ultimately required readmission. These error cases provide critical clinical context, aiding in model refinement and clinical decision-making.

We anticipate that the Random Forest model will demonstrate marginally higher accuracy and F1-score performance relative to Logistic Regression, primarily due to its capability to capture complex, nonlinear feature interactions. However, Logistic Regression is expected to offer superior interpretability, which is highly valuable for clinician trust and decision-making. Our research builds upon established methodologies in hospital readmission prediction, but we further extend their practicality through integration into a user-friendly deployment interface, thereby increasing clinical usability and relevance.

To avoid model overfitting, we will explicitly limit the complexity of our Random Forest model by constraining tree depth and utilizing bootstrapped samples. Similarly, regularization techniques will be applied within Logistic Regression to penalize overly complex models. Conversely, to prevent underfitting, we will ensure sufficient model complexity by carefully selecting and experimenting with relevant features, continuously monitoring performance metrics on validation sets to achieve a balance between model complexity and predictive capability.

3.1.4 Model Deployment

After offline experimentation and metrics comparison, we will choose the model that provides the best combination of high predictive performance and interpretability. This chosen model (or both, for comparative purposes) will be saved in a serialized form so it can be loaded by the Front-End. We also note ethical and societal implications: patient-level data is sensitive, so any real-world deployment would require proper data security, privacy measures, and transparency about model decisions.

3.2. Front-End (Streamlit)

The front-end of our system is constructed using Streamlit, chosen specifically for its capability to quickly develop intuitive, user-friendly interfaces tailored to the needs of hospital staff and data analysts. The user interface is organized into distinct pages to streamline user interaction: a landing page, which briefly describes the project's objectives and introduces the concept of hospital readmission risk; a prediction page, featuring a structured form through which users input relevant patient attributes; and optionally, a visualizations page, offering transparency by displaying summary statistics, histograms, or feature importance charts derived from the underlying machine learning models.

For data entry, the prediction page includes clearly labeled numeric input fields for patient-specific data, such as age, previous hospitalization count, and laboratory test results, as well as dropdown menus for categorical selections like discharge disposition and insurance type. Once data entry is complete, users activate the prediction by clicking a dedicated "Predict" button, which triggers real-time inference using the trained models on the back-end.

The outputs are presented clearly to support clinical decision-making. These include the probability percentage of patient readmission within 30 days, alongside a direct binary classification into "High Risk" or "Low Risk" categories. Additionally, for enhanced interpretability—particularly with models like Random Forest—we include optional explanatory visualizations, such as concise bar charts that illustrate the relative importance of input features.

Integration between the front-end interface and the back-end prediction logic occurs via direct API calls or imported inference functions within Streamlit scripts, enabling seamless real-time prediction capabilities based on user-submitted patient data. Lastly, while the current implementation primarily serves an educational purpose, we explicitly recognize the critical importance of robust security measures—including data encryption, secure storage, and controlled access—to protect patient privacy should real-world deployment occur.

4. Risk & Mitigation

A key challenge for our study involves ensuring data quality and availability. Hospital readmission datasets are often proprietary, incomplete, or contain significant missing values and noise, potentially degrading model performance. To mitigate this, we plan to utilize a well-documented, publicly available dataset (such as those hosted by UCI or Kaggle). We will rigorously apply robust data cleaning and preprocessing methods, including missing value imputation and detailed outlier analysis. In instances of extensive missingness, we intend to reference the original data documentation to guide precise imputation and informed feature engineering decisions.

Another prominent concern is class imbalance, common in clinical datasets where readmitted patients typically represent a small fraction compared to non-readmitted patients. To address this, our evaluation strategy explicitly incorporates metrics such as the F1-Score and ROC-AUC, which appropriately account for skewed distributions. Additionally, we may utilize data-level balancing techniques like random oversampling or Synthetic Minority Oversampling Technique (SMOTE), and algorithm-level approaches such as class weighting in Logistic Regression, to better represent minority classes in model training.

Implementation complexity poses an additional technical challenge, as developing Logistic Regression and Ran-

dom Forest algorithms from scratch is both time-intensive and prone to errors—particularly in intricate processes such as gradient computation and tree splitting. To mitigate this risk, we will systematically divide tasks among team members, perform thorough unit tests, and verify critical algorithmic steps with small, controlled toy examples before scaling implementations to the primary dataset.

Moreover, we recognize the risks of overfitting and underfitting inherent to modeling high-dimensional data. To ensure a balanced model, we will employ regularization techniques within Logistic Regression, set explicit limits on tree depth and complexity within the Random Forest model, and rigorously tune hyperparameters through cross-validation. Conducting detailed ablation studies will further help us isolate and select feature sets contributing to generalizable performance.

In healthcare contexts, interpretability frequently competes with predictive performance, presenting another challenge for our proposed methods. While Logistic Regression inherently provides transparent coefficient interpretations, ensemble methods like Random Forest may sacrifice some interpretability for improved accuracy. To strike a balance, we will present feature importance visualizations for the Random Forest alongside coefficient analyses for Logistic Regression, allowing informed decisions regarding interpretability versus performance trade-offs. If feasible within our timeline, we may consider deploying both models concurrently, providing users with flexible options.

Finally, integrating trained models seamlessly with our Streamlit-based front-end could introduce compatibility or deployment issues. To mitigate these risks, we will establish clearly defined interfaces, leveraging lightweight APIs or direct function calls to reliably exchange data between user inputs and model outputs. By adopting incremental integration strategies, supported by frequent testing, we aim to minimize deployment issues and ensure robust, user-friendly model interaction.

5. Expected Outcomes

Drawing from existing literature, we anticipate achieving improved predictive accuracy in hospital readmission predictions through the implementation of a Random Forest model developed from scratch. Compared to simpler linear methods such as Logistic Regression, Random Forests typically capture intricate, nonlinear relationships among clinical variables more effectively, leading to modest yet measurable improvements in key metrics like accuracy and F1-Score. Nonetheless, we acknowledge the inherent value of interpretability offered by Logistic Regression. Despite potentially lower predictive performance, Logistic Regression provides clear, actionable insights through easily interpretable coefficients, clearly highlighting patient-specific features such as previous admissions, comorbidities, or discharge codes that significantly influence 30-day readmission risk. In healthcare settings, such interpretability is vital, as transparent, explainable models are more readily embraced by clinicians.

To bridge the gap between analytical modeling and practical clinical application, we will integrate our trained models into an intuitive Streamlit front-end, thereby demonstrating a real-time decision-support tool. This system enables hospital administrators and care coordinators to quickly input relevant patient information and instantly receive a personalized readmission risk assessment. Such functionality is expected to streamline clinical workflows, enhance resource allocation, and improve discharge planning.

Finally, our comprehensive end-to-end pipeline—encompassing data preprocessing, custom model training, rigorous evaluation, and deployment via a user-friendly interface—establishes a robust framework for future healthcare-focused machine learning applications. Beyond readmission prediction, this framework can be generalized and adapted for risk assessment tasks in other disease domains. Moreover, future expansions of our approach could incorporate advanced methodologies, such as deep neural network architectures or external data sources including unstructured clinical notes, further enhancing predictive power and clinical applicability.

6. Team Member Contribution

Each team member is expected to contribute to the final project. Task responsibilities must be clearly stated in terms of front- and back-end development.

6.1. Technical Components

In terms of back-end and front-end programming, Aryan will acquire, clean, and preprocess the dataset, addressing any missing values and applying encoding and normalization where needed. He will also implement the Logistic Regression model from scratch, focusing on parameter initialization, cost function derivation, and gradient updates. Omkar will develop the Random Forest algorithm from scratch, including decision tree construction, bootstrapping, and aggregation. He will also lead the hyperparameter tuning and evaluation processes for both models, setting up cross-validation to identify optimal parameters. Neelraj will build and connect the Streamlit front-end to the trained models, managing user inputs, real-time inference requests, and model outputs for display. This ensures a seamless user experience and proper integration of back-end logic.

6.2. Writing Components

Aryan will document the "Data Collection, Exploration Processing" section, detailing the chosen dataset, its cleaning strategy, and justifications for the preprocessing techniques. He will also work on the "Risk Mitigation" portion,

focusing on potential data-related challenges and remedies. Omkar will compose the "Methods and Model Training" and "Model Evaluation" sections, explaining the theoretical underpinnings and hyperparameter configurations for both Logistic Regression and Random Forest, as well as the chosen metrics and experimental setups. Neelraj will produce the "Front-End (Streamlit)" discussion, outlining the user interface, input forms, and prediction outputs. He will likewise complete the "Model Deployment" subsection, describing how models are loaded and used in the web application, along with ethical considerations for patient data. Finally, he will compile and proofread the final document to ensure coherence and clarity across all sections.

References

[1] Isaac Y. Chen, Peter Szolovits, and Marzyeh Ghassemi. Can ai help reduce disparities in general medical and mental health care? In *AMA Journal of Ethics*. American Medical Association, 2019. 2

[2] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. In *Proceedings of the ICML Workshop on Human Interpretability in Machine Learning*. PMLR, 2017. 2

[3] Centers for Medicare & Medicaid Services. Hospital readmissions reduction program. In *Federal Register*. U.S. Department of Health and Human Services, 2012. 2

[4] Stephen F Jencks, Mark V Williams, and Eric A Coleman. Rehospitalizations among patients in the medicare fee-for-service program. In *New England Journal of Medicine*. Massachusetts Medical Society, 2009. 2

[5] Devan Kansagara, Hannah Englander, Anthony Salanitro, Emily R Kagen, Christine Theobald, Mark Freeman, and Sunil Kripalani. Risk prediction models for hospital readmission: a systematic review. In *JAMA*. American Medical Association, 2011. 2

[6] Alvin Rajkomar, Eyal Oren, Kai Chen, Amber M Dai, Narges Hajaj, Michael Hardt, Peter J Liu, X Jeff Liu, Joshua Marcus, Michael Sun, Phillip Sundberg, Harold Yee, Zhengping Zhang, Yining Zhang, Gabriela Flores, Noémie Elhadad, Martin Kosinski, Scott Ludwig, Burney Marks, Karthikeyan Natarajan, David Duvenaud, Nigam H Shah, and Iran Noriega. Scalable and accurate deep learning with electronic health records. In *NPJ Digital Medicine*. Nature Publishing Group, 2018. 2

[7] Karen Shadmi, Nitzan Flaks-Manov, Moshe Hoshen, Orly Goldman, and Ran D Balicer. Predicting 30-day readmissions using electronic health records: A comparison of machine learning methods. In *Journal of Biomedical Informatics*. Elsevier, 2015. 2