Using Free-Text Clinical Notes to Improve Model Performance in Healthcare

By

Daniel Figueiras, Bachelor of Science in Mathematics

A thesis submitted to the Graduate Committee of

Ramapo College of New Jersey in partial fulfillment

of the requirements for the degree of

Master of Science in Data Science

Spring, 2025

Committee Members:

Dr. Osei Tweneboah, Advisor

Dr. Debbie Yuster, Reader

Dr. Amanda Beecher, Reader

COPYRIGHT

Acknowledgments

I would like to thank my thesis advisor, Dr. Tweneboah, for his guidance, as well as my thesis readers, Dr. Amanda Beecher and Dr. Debbie Yuster, for their input throughout this process.

Table of Contents

List of Tables

List of Figures

Abstract

Predictive models in healthcare often rely solely on structured data, missing crucial context contained in free-text clinical notes and thereby limiting accurate outcome prediction. This study quantified the impact of incorporating free-text discharge summaries alongside structured data to improve one-year mortality prediction by evaluating both resampling techniques and Natural Language Processing (NLP) methods.

Using the MIMIC-IV and MIMIC-IV-Note datasets, five machine learning model types were trained with structured data alone versus structured data combined with insights extracted from clinicals notes using four NLP techniques (Bag of Words (BoW), Binary BoW, Term Frequency-Inverse Document Frequency (TF-IDF), and Sentiment Analysis). A hybrid resampling method addressed severe class imbalance. Performance was primarily evaluated using recall due to the nature of outcomes being predicted.

Baseline models, trained using only structured data, obtained poor recall scores (~0.17). Resampling was essential, boosting average recall by ~61.5%. Integrating clinical notes further improved performance. The gradient boosting model trained using TF-IDF features achieved the highest recall (0.779), a 4.6% gain over its baseline after resampling. TF-IDF and BoW were the most effective NLP methods overall. Key features from the best performing model included age and discharge location (from the structure data) and note terms (i.e. CT, disease).

Overall, the inclusion of free-text clinical notes, combined with effective resampling, significantly enhances the performance of healthcare models, resulting in improved identification of high-risk patients and ultimately contributing to better patient care.

I. Introduction

The use of predictive models in healthcare has significantly progressed over the past decades as healthcare data, machine learning models, and computational power have all vastly improved. To fully understand this progress, however, it is important to consider the historical foundations of medical data. Ancient civilizations like Egypt, Greece, and Mesopotamia, recorded early forms of medical records and treatments on papyrus and tablets. Despite the limited technology during this time, the foundations of modern medicine were laid as Hippocrates of Greece created the Hippocratic Corpus, one of the earliest works which methodically recorded medical observations. During the Middle Ages, scholars compiled and further developed medical knowledge and by the 18th century, manual record-keeping of medical observations became a more formal process. This shift allowed for a more systematic approach to documentation of patient information, facilitating patient diagnosis and treatment. This time also saw the field of medical statistics emerge and develop as statistician John Graunt and epidemiologist William Farr used data to identify disease trends, categorize causes of death, and compile national health data. During the mid 19th century, one of the most noteworthy applications of medical statistics occurred as statistician Florence Nightingale demonstrated how health outcomes were affected by sanitation through statistical analysis during the Crimean War. This marked a pivotal moment in exemplifying the power of data in medicine. The introduction of computing power to the medical field in the mid twentieth century marked an even more transformative time. Early computers were used to process and analyze large quantities of data and ultimately led hospitals to digitize patient records. As statistical software developed in the last twentieth centuries, the complexity of the analysis that could be carried out on these now

digitized patient records only increased. By the 2000s, the widespread adoption of electronic health records created centralized repositories of accessible patient records, improving healthcare coordination and paving the way for more advanced analytics. (Olusegun, 2023).

Today, advanced analytical methods are used for predictive modeling in many different areas of healthcare. One common way predictive modeling is used in healthcare is to identify patients with high risk of developing certain diseases. This is done through the creation of machine learning models that predict whether a patient will develop a disease based on a variety of medical and socioeconomic factors. By using these models to identify high risk individuals, targeted intervention and disease prevention measures can be provided to those that need it the most. This ultimately leads to more personalized healthcare and overall improves medical decision making. This is especially important in oncology as frequent hospitalizations and visits to the emergency department can cause increasingly high costs of care while also impacting the quality of life of patients. Identifying high-risk cancer patients at early stages and providing these patients with personalized care can help them avoid expensive visits to the hospital while also improving the care they are provided.

Another way advanced analytical methods are used in healthcare is to assist in the creation of new therapeutic agents. Recent studies have revealed that computational modeling is able to identify which drug targets are most promising for treating various cancers. Researchers have also deployed simulation techniques to mimic the human brain and thus create models that mimic human brain activity. These models can be used to help formulate new therapeutic agents for various diseases. For example, several studies have been conducted using technology like this in order to identify new biomarkers related to Alzeheimer's. Overall, the use of these new advanced technologies allows medical professionals to develop new therapeutic agents more

efficiently (Toma & Wei, 2023). Not only does this allow new treatments to be created and thus more patients to be treated, but it also could reduce some of the incredibly high costs associated with the drug development process. As drug discovery and development becomes more cost-effective, drug costs could decrease and thus create a more affordable healthcare landscape.

Predictive modeling is also used to predict surgery outcomes. For example, a predictive model for patients with epilepsy was able to effectively distinguish patients who no longer had seizures after surgery from those who continued to have them. Similar models were also developed for patients undergoing cataract surgery, neurosurgery, spine surgery, and more. Overall, models like these enable healthcare professionals to offer individualized information to each patient when discussing surgical risks and thus ultimately leads to more informed patients (Toma & Wei, 2023).

There are many more applications of predictive models in healthcare. Despite all their advantages, many models share the same limitation: structured data. Structured data refers to data that is highly organized and preformatted to fall within certain criteria. The organized nature of structured data is one of its greatest strengths as it allows the data to be easily handled and analyzed. Because of this, most electronic health records come in the form of structured data and thus structured data is often the sole source of information used to train predictive models in healthcare. However, the rigid format of structured data often excludes valuable contextual details. As mentioned prior, in order to ensure structured data is organized, it must be preformatted to fall within a certain criteria. What this means is that structured data must be broken up into certain variables. An example entry from structured data can be found in Table 1.1 below.

Table 1.1. Example entry from structured data

Age	Gender	Diagnosis	Ejection Fraction	Blood Pressure	Sodium Level	Discharge Medications	Length of Stay
72	Female	Congestive Heart Failure (ICD-10: 150.9)	35%	140/90	134	Beta-blocker, ACE inhibitor, Diuretic	4 days

This example entry contains standard patient information that is typically found in an electronic health record. As shown, the variables this dataset contains are age, gender, diagnosis, ejection fraction, blood pressure, sodium level, discharge medications, and length of stay. All of these variables contain crucial information regarding this patient's situation but context is clearly lacking. Context is very difficult to capture in structured data due to its dynamic nature. For example, the data in Table 1.1 reveals that the patient is not doing well given their low ejection fraction, elevated blood pressure, and low sodium levels. This would lead one to believe that this patient is at high risk for hospital readmission. However, this may not be so true when considering the context. For example, if this patient has a strong support system of friends and family members who can help them, they would likely be less at risk for readmission. Unfortunately, this information is very difficult to capture within a variable and thus is often left out of predictive models.

Fortunately, there are medical data sources that do capture information like this called clinical notes, or clinical documentation. Clinical notes typically contain summaries of medical observations that arise from patient care and serve multiple purposes, such as creating a record of a patient's history and medical findings, recounting care and procedures in case of any future arbitration, justifying the amount of reimbursement for provided services, and determining the quality of care a patient received (Rosenbloom et al., 2010). Clinical notes arise from many

different aspects of healthcare, such as outpatient visits, inpatient admission, inpatient discharge, and medical procedure protocols and results. Studies have found that clinical notes containing natural prose are more reliable for identifying patients with certain diseases, easier for healthcare personnel to understand, and overall more accurate (Rosenbloom et al., 2011). Evidently, clinical notes contain valuable information that is lacking in structured data. As a result, including clinical notes in the data used to train predictive models could improve the predictive power of such models and thus improve medical outcomes.

This study aims to analyze the effect that free-text clinical notes have on the predictive power of models in healthcare. To achieve this, various predictive models will first be developed using only structured data to establish baseline performance and identify which models perform best. These same models will then be retrained using the same structured data combined with free-text clinical notes, and their performances will be subsequently analyzed to determine the added value of the clinical notes. To extract insights from the free-text clinical notes so that they can be used to train the models, this study will explore multiple natural language processing (NLP) techniques. Consequently, the results of this study will also provide an analysis of which natural language processing methods contribute most to high model performance.

The remainder of this paper is structured as follows: It begins with a literature review and discussion of prior work completed on this subject. Next, it examines the ethical considerations of working with medical data and predicting medical outcomes. The Methodology section then outlines the process used to address the research question. Finally, results and implications are presented, followed by a conclusion summarizing key findings.

II. Background

Literature Review

While structured data has traditionally been the sole source used to train predictive models in medicine, researchers have increasingly begun utilizing free-text clinical notes to improve model performance. This chapter will review some of the prior research completed that leverages clinical notes in their predictive models. It will analyze different methodologies used with emphasis on different model types, natural language processing techniques, and model evaluation methods. Additionally, it will examine the model performances obtained using different data types. Overall, this chapter aims to summarize the progress made in the research of this subject while also highlighting the gap this study will fill.

Song et al. conducted a study aimed at improving risk prediction for hospitalization and emergency department visits in home health care patients. This study examined the performance of five different model types - logistic regression, random forest, Bayesian network, support vector machine, and naive Bayes - trained using only structured data and a combination of both structured data and free-text clinical notes. Two different natural language processing techniques were employed to extract information from the clinical notes: convolutional neural networks to label each clinical note as either concerning or not concerning and a rule-based approach which identified a set of predefined risk factors within each clinical note. The models were trained using ten-fold cross validation, with SMOTE applied to the data to address the class imbalance. This study ultimately found that the inclusion of the clinical notes using both natural language processing methods led to the best model performance. Overall, adding in the clinical notes to

the model training process on average improved each model's F-score by roughly 17% and area under the precision-recall curve (AUPRC) by roughly 18% (Song et al., 2022).

Garriga et al. explored the integration of free-text clinical notes with structured data to predict the probability of a mental health relapse within a period of twenty-eight days. Following the use of BERT to process the unstructured data, four different models were trained. A structured XGBoost model and structured deep neural network were fit to only the structured data. Correspondingly, an unstructured deep neural network was fit to only the unstructured data and a hybrid deep neural network was fit to both data types. Additionally, an ensemble deep neural network was trained. This study ultimately found that the models trained on both the structured and unstructured data performed better than the models trained on just one data source, with the ensemble model having the best performance of all models based on AUPRC and area under the receiver operating characteristic curve (AUROC). This study also conducted an analysis of how many clinical notes were needed in order for them to add value to predictive modeling by dividing the cohort of patients into six subsets based on the proportion of weeks in which they had a clinical note written while receiving care. This analysis showed that having as little as just 10% percent of weeks with at least one clinical note resulted in improved model performance, showcasing the value that even small amounts of clinical notes can add (Garriga et al., 2023).

Huang et al. investigated the use of clinical notes when predicting in-hospital mortality or an ICU length of stay greater than seven days. The purpose of this study was not only to examine the performance of models trained purely on clinical notes, but also to analyze if performance differed when using nursing notes or physician notes. The dataset included patients that were at least eighteen years old and had ICU stays greater than two days. To extract insights from the

clinical notes, the bag of words method was used and the top 3,000 most frequent words were selected. Following this, ten percent of the dataset was separated to use as a holdout test set and the remaining ninety percent was randomly divided into training and testing sets five times. To show consistency among results, penalized logistic regression, random forest, and gradient boosting models were trained and tested. Using AUC to measure model performance, all three model types found that nursing notes contributed to better performance than physician notes, though the best model performance was obtained when using both the nursing notes and physician notes. This study also included a variable importance analysis (Huang et al., 2021).

Gao et al. conducted a study during which free-text admission notes and structured data were used in the creation of predictive models which aimed to predict the mortality of patients with heart failure. To do this, all first time ICU admissions of patients with heart failure above the age of sixteen were extracted from multiple data sources and split into four subsets used to train and evaluate the models. The free-text clinical notes were also extracted and separated into five different categories. The models were created using a supervised multimodal deep learning framework and a pretrained BERT module was used to process the clinical notes. This study ultimately found that the models trained using both the structured data and all five categories of the clinical notes outperformed the models trained using only one data source - the structured data or the clinical notes. Model performance was evaluated using AUROC, F1 score, and AUPRC. This study also deployed feature analysis methods to further interpret the model outcomes and identify the most influential features (Gao et al., 2024).

While prior studies have exemplified the value of using free-text clinical notes in the predictive modeling of healthcare outcomes, they often consider a limited number of different model types and natural language processing techniques. This study will take a more

comprehensive approach in that it will apply and analyze five different modeling approaches and four different natural language processing techniques. Furthermore, this study will provide a unique analysis of resampling techniques. Resampling is often crucial for training predictive models in healthcare due to the unbalanced nature of medical outcomes. While some prior studies have mentioned the use of specific resampling techniques, this study will test and compare six different resampling techniques. By including these additions to the methodology, this study aims to provide results that are more robust than prior work and thus contribute to an improved framework for using free-text clinical notes in predicting medical outcomes.

Methods

To analyze the effect of free-text clinical notes on models in healthcare, this study leveraged many different techniques and metrics. This section will discuss all model types, model evaluation metrics, resampling techniques, natural language processing techniques, and other associated techniques applied in this study.

Modeling

The selection of modelling techniques used in this study was made strategically to include a mix of simple and complex models, including some ensemble methods. In addition to describing each of these techniques, this section will also include a description of one-hot encoding, a data transformation technique vital for certain model types. All techniques described in this section are standard practice for classification tasks like that of this study. Note that the classification task in this study is binary.

Logistic Regression

Logistic regression is one of the most widely used models for classification tasks. Based on the provided explanatory variables, this kind of model outputs the probability p(X) that the response variable (Y) belongs to a certain class using the logistic function, shown in Equation 2.1.

$$\text{Logistic Function: } p(X) = \frac{e^{\beta_0+\beta_1X_1+\ldots+\beta_nX_n}}{1+e^{\beta_0+\beta_1X_1+\ldots+\beta_nX_n}} \quad \textbf{(2.1)}$$

To determine the coefficients (β_0, β_1, . . ., β_n), logistic regression uses maximum likelihood estimation, a fairly general approach often used for non-linear models. This method finds the coefficient values which result in probabilities that most closely resemble the given data, assigning a probability as close to one as possible for the positive class and as close to zero as possible for the negative class (James et al., 2023). Because logistic regression outputs probabilities, one must determine a threshold for assigning each probability to each class. While this can vary depending on the outcomes being modeled, it is common practice to use 0.5 as the threshold. That is, any possibility above 0.5 will be assigned to the positive class and any below 0.5 will be assigned the negative class. All logistic regression models created in this study use 0.5 as the threshold value.

Decision Trees

A decision tree is a simple and interpretable type of model that makes predictions by segmenting the predictor space into multiple regions based on certain decisions located at decision nodes. While decision trees can be used for both regression and classification tasks, this study leveraged the classification version due to the nature of outcomes being predicted. In a classification decision tree, each observation is predicted to belong to the most commonly occurring class of the training dataset within the region it lies. These regions are optimally determined by selecting the decision nodes that best split the data into separate groups. This is

commonly determined using either the Gini index or entropy criterion (the decision trees in this study use Gini index) (James et al., 2023). Figure 2.1 below displays an illustrative decision tree used to predict hospital admittance.

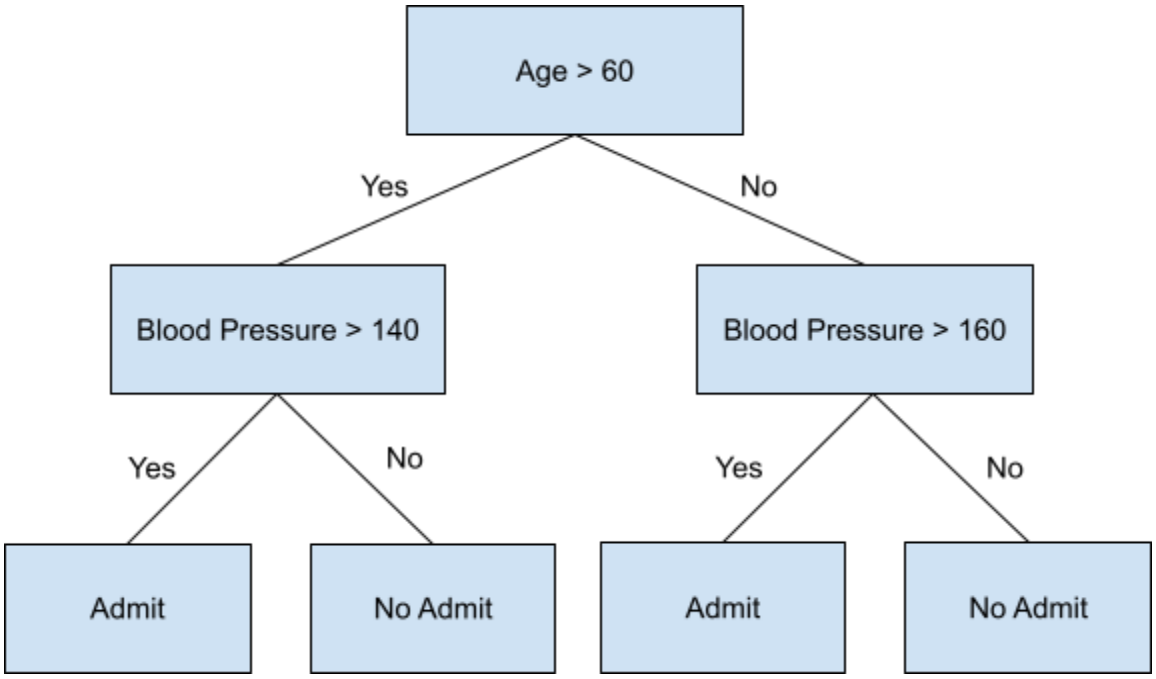


Figure 2.1. *Example decision tree used to predict hospital admittance - example was created for illustrative purposes and is not based on the data used in this study.*

The illustrative decision tree above consists of three decision nodes. The first decision node, located at the top, separates patients above and below age sixty. These two groups then have their own respective decision nodes related to blood pressure. Using this tree, a sixty five year old patient with a blood pressure of 120 would be predicted to not be admitted to the hospital as they would fall within the left region of the tree due to being older than sixty five and would then take the right path due to having a blood pressure below 140.

Random Forests

A random forest is an ensemble method that builds multiple decision trees using bootstrapped training data samples. When building each decision tree, only a subset of predictors

is considered so that each tree contains meaningful differences. Typically, the size of the subset of predictors used in each tree is roughly equal to the square root of the total number of predictors, meaning that each tree will not even consider the majority of the predictors in the overall training set. This is especially important in cases where either one feature is significantly more important than all others or where many predictors are very correlated to each other. Once all decision trees are fit, a random forest makes predictions by aggregating the results of each decision tree. In the context of classification, this means that the most common prediction of all trees is typically taken as the final prediction of the forest. In general, random forests avoid overfitting when a sufficiently large number of decision trees are created (James et al., 2023).

Gradient Boosting

Boosting is another ensemble method that can be applied to decision trees. In this method, decision trees are created sequentially with each tree using information from the tree that came before it. More specifically, with each decision tree, a new decision tree is fit to the residuals (the difference between the predicted values and actuals), rather than the response variable. This allows the model to improve gradually, improving in areas where the model does not perform well (James et al., 2023). Gradient boosting is a specific kind of boosting that fits each new decision tree to the negative gradient of the loss function, rather than simply the residuals.

Neural Networks

Artificial neural networks (ANN), commonly referred to as neural networks, are a type of machine learning model inspired by the function of the human brain. The human brain consists of billions of neurons, all of which receive input signals from stimulation or other neurons. These signals are then processed and transmitted to the output terminal where the output is then sent to

other neurons or other parts of the body to perform actions. Artificial neural networks perform in a similar manner wherein the predictor variables act as the input signals and are received by input nodes. Each feature is then weighted by importance and processed so that the output node can apply the activation function, combining information from the input nodes. These nodes are arranged in layers, with more layers resulting in a more complex model capable of handling more complex tasks (Zhang, 2016). An illustrative example of a neural network can be found in Figure 2.2 below.

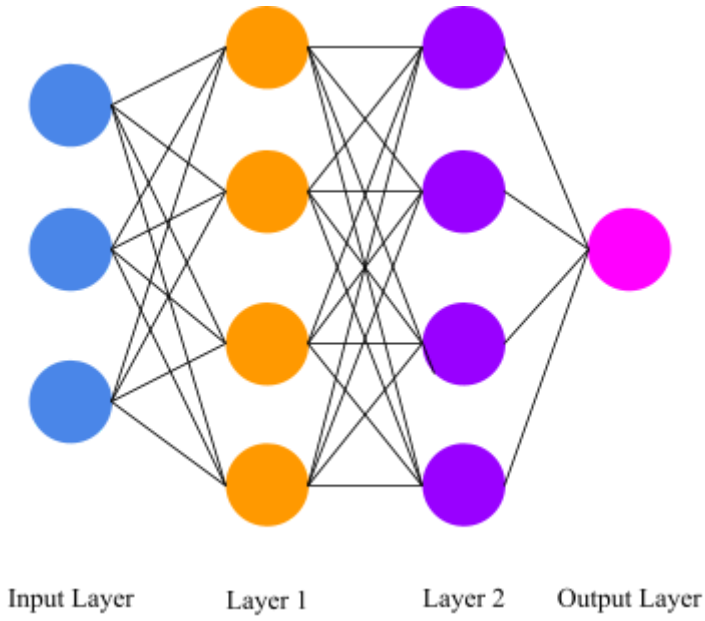


Figure 2.2. *Example neural network.*

In the above neural network, there are three input nodes followed by two layers containing four nodes each. The number of nodes in each layer and the number of layers varies from model to model.

When training a neural network, the model aims to optimize the weights of each feature to the point at which the model's predictions are most accurate. Through this process, neural

networks are able to identify underlying patterns in data better than most standard modeling techniques, making them well-suited for complex tasks (Zhang, 2016).

One-hot encoding

Of the models described above, logistic regression and neural networks require all training data to be quantitative. As a result, all categorical data must be converted to numerical prior to training these models. One very common method for making this conversion is one-hot encoding. Consider the example shown in Table 2.1 below.

Table 2.1. Example of one-hot encoding

ID	Sex	Age
1	M	34
2	F	53
3	F	28

ID	Sex_M	Sex_F	Age
1	1	0	34
2	0	1	53
3	0	1	28

Above, the table on the left contains one categorical feature, sex, containing two unique categories, M and F. The result of applying one hot encoding is shown on the table on the right. Evidently, one-hot encoding creates one column for each category of a categorical feature, with each new column using binary to indicate the original value. For example, because the first row has the value M for sex, sex_M contains a one and sex_F contains a zero after one-hot encoding. The same logic can be applied for both remaining rows in the table. A best practice for applying one-hot encoding prior to modeling is to remove one of the newly created columns to avoid multicollinearity. This is evident through the example in Table 2.1 as having both the sex_M and sex_F columns is redundant because the value of one column can be immediately deduced from the other. For example, if the value of the sex_M column is one, the value of the sex_F column will always be zero.

Model Evaluation Metrics

All models created in this study were evaluated using the following metrics: accuracy, precision, recall, F1 score, and area under the ROC curve (AUROC). This section will define each of these metrics as well as k-fold cross validation, a model evaluation technique which is used to assess model generalization.

K-Fold Cross Validation

K-fold cross validation is a model evaluation technique which splits the modeling dataset into k subsets of equal size and subsequently runs k experiments (Typically, k is chosen to be either five or ten. All models in this study use k equal to 5). For each experiment, all subsets but one are used to train the model and the remaining subset is used for testing. After all experiments are run, the average performance across all tests - measured using model evaluation metrics - is computed to provide a more stable and accurate idea of model performance. Not only does this technique more reliably assess model generalizability, but it also reduces the risk of overfitting by running multiple experiments.

Accuracy

Accuracy is a simple model evaluation metric that measures the amount of correct predictions made by a model. Accuracy is calculated using the below formula.

$$Accuracy = \frac{\#\ Correct\ predictions}{\#\ Total\ predictions} = \frac{TP + TN}{TP + TN + FP + FN} \quad \textbf{(2.2)}$$

In the above formula, TP and TN denote true positive and true negative, respectively. A true positive represents a case where a model correctly predicts the positive class and a true negative represents a case where a model correctly predicts the negative class. Analogously, FP and FN denote false positive and false negative, respectively. A false positive represents a case where a

model incorrectly predicts the positive class and a false negative represents the case where a model incorrectly predicts the negative class.

Precision

Precision is a model evaluation metric which measures how accurate the positive predictions of a model are. It is calculated using the formula shown below.

$$Precision = \frac{TP}{TP+FP} \quad \textbf{(2.3)}$$

This metric is typically used when the cost of false positive predictions is high.

Recall

Recall is a model evaluation metric which measures how well a model predicts the actual positive class. It is calculated using the formula shown below.

$$Recall = \frac{TP}{TP+FN} \quad \textbf{(2.4)}$$

This metric is typically used when the cost of false negatives is high.

F1 Score

F1 score is a model evaluation metric that takes into account both precision and recall. It can be calculated using the formula shown below.

$$F1\ Score = 2 \cdot \frac{Precision \cdot Recall}{Precision+Recall} \quad \textbf{(2.5)}$$

F1 score is typically used when both the cost of false positives and false negatives is considerable.

AUROC

The receiver operating characteristic curve (ROC) is a graph that plots the relationship between true positive rate (recall) and false positive rate ($\frac{FP}{FP+TN}$) across different classification thresholds. The area under the ROC curve (AUROC) quantifies this relationship as a value

between zero and one, with a score of one indicating a perfect model and a score of .5 indicating a random model. In general, higher values indicate stronger model performance. AUROC is typically used with imbalanced data, making it particularly useful in healthcare settings.

Resampling

Resampling is the process of altering a dataset in order to address class imbalance with the end goal of improving model performance. Resampling techniques can be categorized into two main categories: oversampling and undersampling. Oversampling is the process of increasing the number of data points in the minority class to increase its size relative to the majority class. Conversely, undersampling is the process of decreasing the number of data points in the majority class to decrease its size relative to the minority class. The method by which data points are added or removed depends on the resampling technique applied. This section will define all such techniques applied in this study.

Random Oversampling

Random oversampling is an oversampling technique which randomly creates copies of data entries from the minority class until the size of the minority class reaches a certain threshold. This increases the size of the minority class, thus reducing class imbalance.

SMOTE

Synthetic Minority Oversampling Technique (SMOTE) is an oversampling technique that creates synthetic data entries from the minority class until the size of the minority class reaches a certain threshold. SMOTE creates synthetic samples by using k-nearest neighbors. The first step is to take the difference between the data point of interest (a randomly selected data point from the minority class) and one of its k nearest neighbors (also randomly chosen). This difference is then multiplied by a random value between zero and one, and then added to the data point of

interest. This creates a random point in between the data point of interest and its nearest neighbor, ultimately reducing class imbalance and leading to more general decision regions of the minority class (Chawla et al., 2002).

Random Undersampling

Random undersampling is an undersampling technique which randomly removes data entries from the majority class until the size of the majority class reaches a certain threshold. This decreases the size of the majority class, thus reducing class imbalance.

Edited Nearest Neighbors

Edited nearest neighbors is a unique resampling technique in which data entries from both the majority and minority class are removed. For this technique, the k nearest neighbors of each data point are found and the majority class of each group of neighbors is assessed. For each data point, if the majority class of its neighbors differs from the class of the data point, the data point is removed. This helps remove noise and sharpen the decision boundaries within a dataset, often leading to improved model performance (Wilson, 1972).

Hybrid Techniques

Resampling techniques can be combined to form hybrid resampling techniques that can lead to further improved model performance. This is often done with one oversampling and one undersampling technique. In general, when combining an undersampling and oversampling technique, it is best to apply the undersampling technique first as, depending on the techniques applied, it may prevent unnecessary growth in data size, improve the decision boundary between the minority and majority classes, and reduce redundant majority samples before oversampling, allowing oversampling to be applied to a cleaner and more balanced dataset. In this study, both

random oversampling and SMOTE will be individually combined with random undersampling to form two hybrid techniques.

Natural Language Processing

Natural language processing (NLP) enables computers to understand human language. For this study, natural language processing will be applied to extract insights from free-text clinical notes so that they can be used for modeling. This is crucial for this study as all model types tested require all data to be entered in a structured form. As a result, various NLP techniques will be applied and their output will be formatted in structured form so that it can be used to train models. This section will discuss the four NLP techniques analyzed in this study.

Bag of Words

Bag of words is a simple NLP technique which counts the frequency of each word in a group of documents or notes. For example, consider the following two notes:

Note 1: Patient has fractured left arm and right arm.

Note 2: Patient has fractured leg.

Table 2.2 below displays the structured output of applying bag of words to these notes.

Table 2.2. Example of bag of words

	patient	has	fractured	left	arm	and	right	leg
Note 1	1	1	1	1	2	1	1	0
Note 2	1	1	1	0	0	0	0	1

As shown above, bag of words creates a table in which each unique word from all notes has its own column and each row keeps track of how many times each word appears in each note. For instance, the row for note one contains a two in the *arm* column because that word appears twice in the note. Meanwhile, it contains a zero in the *leg* column as that word does not appear in the

note at all. When applying the bag of words method for analysis, it is typical to remove highly frequent but non-informative words (called stop words), like “and,” “the”, or “is.” If removing stop words from the given example, the *and* column would be removed from the table.

Binary Bag of Words

Binary bag of words is an NLP technique very similar to standard bag of words, described above. Instead of calculating the frequency of each word in each note, binary bag of words keeps track of whether or not each word exists in each note. Table 2.3 below displays the output of applying binary bag of words on the example described above.

Table 2.3. Example of binary bag of words

	patient	has	fractured	left	arm	and	right	leg
Note 1	1	1	1	1	1	1	1	0
Note 2	1	1	1	0	0	0	0	1

As shown in the above table, the output of binary bag of words is structurally identical to that of bag of words. The one key difference between Table 2.3 and Table 2.2 is that Table 2.3 contains a one instead of a two in the *arm* column for the first row. This is due to the fact that binary bag of words does not count the frequency of the word “arm” in note one, but rather uses the number one to indicate that the word “arm” exists in the note. Like with standard bag of words, it is also common practice to remove stop words when applying this technique.

Term Frequency-Inverse Document Frequency (TF-IDF)

TF-IDF is an NLP technique which measures how important each word is to a document, relative to a group of documents. This is done using the formula shown below, where *t* represents a term and *d* represents a document.

$$TFIDF(t,d) = \frac{count\ of\ term\ t\ in\ document\ d}{total\ terms\ in\ document\ d} \times log(\frac{N}{DF(t)+1}) \quad \textbf{(2.6)}$$

,where *N* represents the total number of documents and *DF(t)* is the number of documents containing term *t*. For example, consider a collection of 100 clinical notes in which the word "patient" exists in every note. If applying TF-IDF to the only note that contains the word "hypertension" (one time), the TF-IDF score of hypertension would be much higher than that of the word "patient." Essentially, the word "hypertension" would be deemed as very important in this note as it is the only note containing the word. Despite the fact that the word "patient" appears the same amount of times as the word "hypertension" in that note, the word "patient" would be deemed less important because it appears in every note. Although the TF-IDF formula should apply low scores to common words, it is still considered a best practice to remove stop words when applying the technique.

Sentiment Analysis

Sentiment analysis is an NLP technique that determines the overall sentiment in a text. Sentiment analysis can be conducted in a variety of ways and can result in different kinds of output. The method of sentiment analysis used in this study, acquired from the sentiment module of the NLTK (Natural Language Toolkit) Python library, results in a numeric output value between negative one and positive one, where positive one indicates a positive sentiment, negative one indicates a negative sentiment, and zero indicates neutrality.

III. Ethical Considerations

While the purpose of this study is to examine the use of free-text clinical notes in predictive modeling through the use of various resampling, modeling, and natural language processing techniques - rather than developing models to be applied in real world medical settings - it is still important to discuss the ethical implications of predictive modeling in healthcare. A significant portion of the ethical concerns regarding modeling in healthcare arises from the use of medical data. The three main concerns within this domain are privacy, consent, and bias. Privacy is a major concern because medical data often includes personally identifiable information (PII) which, if obtained by malicious individuals, can be used to commit insurance fraud or identity theft. Because of this, it is critical to properly anonymize the medical data used to train models so that individuals cannot be identified, thus ensuring their sensitive information is safe (Uwinama et al., 2023).

Informed consent has long been a key moral principle in medicine as it is important for patients to fully understand the risks and benefits associated with their medical care so that they can make informed decisions. However, the introduction of machine learning and big data has added some new complexities to this traditional idea as it is often difficult for patients to comprehend how their data will be used. As a result, it is crucial to clearly explain to patients how their data will be used and the implications of such uses before obtaining their consent (Uwinama et al., 2023). The one caveat to this is data de-identification as oftentimes consent is not required when data is fully de-identified.

Finally, bias is also an incredibly relevant topic as the use of biased data in the training of machine learning models can lead to substandard model performance - an issue that can be

especially detrimental given the severity of some medical scenarios. Bias in medical data can appear in several ways. One of the more straightforward forms of bias in data is imbalanced sample sizes. This is especially relevant for medical data as many medical outcomes are relatively uncommon. Additionally, medical data often contains imbalance in some demographic characteristics, like race, as the locations from which data is collected often does not represent the distribution of the general population due to a wide variety of reasons. Another common form of bias in medical data arises from missing data, which is often missing due to nonrandom reasons. For example, individuals of low socioeconomic status often receive less treatment for certain diseases, thus causing there to be less available data for this group of people. More broadly, if and how a patient seeks medical treatment varies among different socioeconomic groups, creating systemic gaps in data that ultimately carry over to predictive models. Bias in data can also be caused by data labels and misclassification. Of course, misclassification, like diagnosing a patient with hypertension when they are within the normal range of blood pressure, can have a negative effect on predictive modeling as the data is not accurate. However, data labels themselves, even when classifying a patient as intended, can still lead to bias because they often do not fully capture the complexities of medical outcomes due to the subjective nature of diagnoses made by each individual care provider. As a result, the way data labels are created can have a significant effect on how well a model understands real world medical settings (Cross et al., 2024).

Ethical concerns regarding modeling in healthcare also arise from the use of the models themselves. For example, the use of predictive models in medicine raises ethical questions regarding trustworthy communication and the relationship between patients and clinicians as patients will no longer know if the medical recommendations given to them by their healthcare

providers are simply a professional medical opinion or if they were formed using predictive models. This can ultimately lead to a loss of trust between patients and healthcare personnel. More ethical concerns stem from autonomy and how to define responsibility for the outcomes of predictive models (Petersson et al., 2023). For example, consider a model that predicts patient mortality within some time frame. A model like this could be used to determine which patients need the most immediate care. However, there are serious implications of a model like this getting one of its predictions wrong. If the model were to predict that a patient had a very low probability of death in the near future when that patient was actually high risk, it could cause medical professionals to not prioritize the care of this individual and thus put them in a dangerous situation. Conversely, if the model were to predict that a patient had a very high probability of death when the patient was not actually high risk, it could lead to the expenditure of resources that were not needed for that patient and could have been used for someone else. It is also important to consider the outcomes of predictive models when they are correct. For example, considering this same model, if it were to predict that a patient has a very high probability of mortality, it could cause some to believe that resources should not be spent on that patient as their situation is too severe. Overall, the predictions made by models in healthcare can have drastic consequences and someone needs to be held accountable for those decisions even though they are not made by humans. Because of this, it is imperative to emphasize that predictive models should simply be used as tools to assist medical professionals and not as a replacement for human judgement.

IV. Methodology

Dataset

Version 3.1 of the Medical Information Mart for Intensive Care IV (MIMIC-IV) dataset was used to conduct this study. MIMIC-IV contains a large collection of de-identified data from adult patients admitted to the emergency department or intensive care unit (ICU) at the Beth Israel Deaconess Medical Center (BIDMC) located in Boston, Massachusetts. The dataset contains records from over 200,000 patients admitted to the emergency department and more than 65,000 patients admitted to the ICU. MIMIC-IV is derived from the BIDMC's custom electronic health records and a clinical information system specific to the ICU. This dataset was formed through a multi-step process, beginning with data acquisition. During data acquisition, a master patient list was developed for all patients admitted to the emergency department or ICU between 2008 and 2022 to keep track of all patients within the dataset. The data was then reorganized to enable more efficient retrospective data analysis by removing audit trails, denormalizing tables, and restructuring the data into fewer tables. The final step in the creation of the dataset was de-identification. All patient identifiers as specified by the Health Insurance Portability and Accountability Act (HIPAA) were removed and replaced with random ciphers, while all free-text data was anonymized using a free-text de-identification algorithm. The dates and times for each hospital visit were also randomly shifted into the future to further de-identify the data while ensuring the data for each individual patient remained temporally ordered. So, for example, if a patient had two procedures done four hours apart, this time difference would still be reflected in the data. However, two distinct patients admitted to the emergency department on the same day would likely have two distinct days of admission in the data (Johnson et al., 2024).

The MIMIC-IV dataset is organized into two modules: the *hosp* module, sourced from EHRs, and the *ICU* module, sourced from the ICU information system. The *hosp* module primarily contains data collected during the hospital stays, though some information, such as outpatient lab tests, originate from outside hospital admissions. This module consists of twenty-two unique tables and includes information like patient demographics, hospitalizations, lab measurements, provider orders, medication administration, and more. Meanwhile, the *ICU* module contains data more specific to stays in the intensive care unit. This module is separated into nine tables and contains information regarding intravenous and fluid inputs, patient outputs, medical procedures, and other charted information (Johnson et al., 2024).

This study also leveraged the MIMIC-IV-Note dataset, a companion to the MIMIC-IV dataset which contains de-identified free-text clinical notes. This study will make use of the over 300,000 discharge summaries that are present for roughly 150,000 patients admitted to the emergency department or hospital at BIDMC. The same inclusion criteria defined for the MIMIC-IV dataset were also applied for this notes dataset. De-identification was performed using both a neural network trained for de-identification and a custom rule-based approach, achieving highly accurate removal of protected health information (Johnson et al., 2023). An example discharge summary from the MIMIC-IV-Note dataset can be found in Figure 4.1 below.

Name: ___ Unit No: ___ Admission Date: ___ Discharge Date: ___ Date of Birth: ___ Sex: F Service: UROLOGY Allergies: Patient recorded as having No Known Allergies to Drugs Attending: ___. Chief Complaint:renal mass Major Surgical or Invasive Procedure:right laparascopic radical nephrectomy- Dr. ___, Dr. ___ ___ History of Present Illness:___ y/o healthy female with incidental finding of right renal mass suspicious for RCC following MRI on ___. Past Medical History:PMH: nonspecific right axis deviation PSH- cesarean sectionALL-NKDA Social History:___Family History:no history of RCC Pertinent Results:___ 07:15AM BLOOD WBC-7.6 RBC-3.82* Hgb-11.9* Hct-33.8* MCV-89 MCH-31.2 MCHC-35.2* RDW-12.8 Plt _____ 07:15AM BLOOD Glucose-150* UreaN-10 Creat-0.9 Na-138 K-3.8 Cl-104 HCO3-27 AnGap-11 Brief Hospital Course:Patient was admitted to Urology after undergoing laparoscopic right radical nephrectomy. No concerning intraoperative events occurred; please see dictated operative note for details. The patient received perioperative antibiotic prophylaxis. The patient was transferred to the floor from the PACU in stable condition. On POD0, pain was well controlled on PCA, hydrated for urine output >30cc/hour, provided with pneumoboots and incentive spirometry for prophylaxis, and ambulated once. On POD1,foley was removed without difficulty, basic metabolic panel and complete blood count were checked, pain control was transitioned from PCA to oral analgesics, diet was advanced to a clears/toast and crackers diet. On POD2, diet was advanced as tolerated. The remainder of the hospital course was relatively unremarkable. The patient was discharged in stable condition, eating well, ambulating independently, voiding without difficulty, and with pain control on oral analgesics. On exam, incision was clean, dry, and intact, with no evidence of hematoma collection or infection. The patient was given explicit instructions to follow-up in clinic with ___ in 3 weeks. Medications on Admission:none Discharge Medications:1. Hydrocodone-Acetaminophen ___ mg Tablet Sig: ___ Tablets PO Q6H (every 6 hours) as needed for break through pain only (score >4) .Disp:*60 Tablet(s)* Refills:*0*2. Docusate Sodium 100 mg Capsule Sig: One (1) Capsule PO BID (2 times a day).Disp:*60 Capsule(s)* Refills:*2* Discharge Disposition:Home Discharge Diagnosis:renal cell carcinoma Discharge Condition:stable Discharge Instructions:-You may shower but do not bathe, swim or immerse your incision.-Do not eat constipating foods for ___ weeks, drink plenty of fluids-Do not lift anything heavier than a phone book (10 pounds) or drive until you are seen by your Urologist in follow-up-Tylenol should be used as your first line pain medication. If your pain is not well controlled on Tylenol you have been prescribed a narcotic pain medication. Use in place of Tylenol. Do not exceed 4 gms of Tylenol in total daily-Do not drive or drink alcohol while taking narcotics-Resume all of your home medications, except hold NSAID (aspirin, advil, motrin, ibuprofen) until you see your urologist in follow-up-If you have fevers > 101.5 F, vomiting, or increased redness, swelling, or discharge from your incision, call your doctor or go to the nearest ER-Call Dr. ___ to set up follow-up appointment and if you have any urological questions. ___ Followup Instructions:___

Figure 4.1. *An example discharge summary from the MIMIC-IV-Note dataset.*

As shown in the above figure, the discharge summary contains free-text that follows standard sentence structure at many points throughout the note. It also seems that it follows a predefined structure in that it begins a general set of information about the patient, including name, unit number, admission date, discharge date, date of birth, and sex of the patient. After this, the note discusses the patient's reason for attending the hospital and summarizes the care she received while admitted. The note then ends with instructions given to the patient at discharge.

Data Preparation & Modeling

Following the acquisition of the MIMIC data sources, I began preparing the data for analysis using Python. As discussed prior, the MIMIC data sources contain large amounts of data, most of which were not required for this study. Therefore, I first identified which features

from the structured data to use for modeling by reviewing the content within each table of the MIMIC-IV dataset. I chose the *admissions* table from the *hosp* module as the starting point as it contains information about all 546,028 hospitalizations across the dataset. Each entry in this table represents a unique hospitalization and contains sixteen variables, including unique patient and admission identification codes, admission and discharge information, and demographic information. Table 4.1 below displays a sample of the *admissions* table.

Table 4.1. A subset of the *admissions* table

Subject ID	HADM ID	Admit Time	Disch Time	Admission Type	Admission Location	Discharge Location	Insurance	Race	Hospital Expire Flag
10000032	22595853	2180-05-06 22:23:00	2180-05-07 17:15:00	URGENT	TRANSFER FROM HOSPITAL	HOME	Medicaid	WHITE	0
10000032	22841357	2180-06-27 18:27:00	2180-06-27 18:49:00	EW EMER.	EMERGENCY ROOM	HOME	Medicaid	WHITE	0
10000032	25742920	2180-08-05 23:44:00	2180-08-07 17:50:00	EW EMER.	EMERGENCY ROOM	HOSPICE	Medicaid	WHITE	0
10000032	29079034	2180-07-23 12:35:00	2180-07-25 17:55:00	EW EMER.	EMERGENCY ROOM	HOME	Medicaid	WHITE	0
10000068	25022803	2160-03-03 23:16:00	2160-03-04 6:26:00	EU OBSERVATION	EMERGENCY ROOM	NaN	NaN	WHITE	0

Using the admission time and discharge time variables shown in the above table, I created a new variable to measure the length of stay of each patient by subtracting the discharge time from the admittance time, measuring the time in days. In order to carry out this calculation, I first converted both the time variables into datetime objects.

Next, I turned to the *patients* table to extract more demographic information about each patient. More specifically, I extracted both age and gender from the *patients* table, using subject ID as the primary key. In addition to age and gender, I also extracted the date of death of each patient. As part of the de-identification process, date of death is only contained within the MIMIC-IV dataset if the patient's death occurred within one year of their most recent hospital discharge. As a result, this variable allowed me to define one-year mortality by finding the

difference between the date of death and discharge time. To do this, I also had to convert the date of death variable into a datetime object. I called the resulting variable death_flag as it was binary (1 indicating the patient died within one year of their discharge) and decided to use this as my dependent variable when modeling.

After defining this variable, I then reviewed the *icustays* table in the *ICU* module which contains information about 94,458 unique ICU stays. Because the *ICU* module contains data on a relatively small sample of the overall patients, I decided not to use any of its variables to keep my model as general as possible. However, I did create a new binary variable based on the *ICU* module to indicate whether or not each patient was admitted to the ICU during their hospital stay. This was done by using the hospital admission ID (hadm_id) as the primary key. I then created the final variable, previous hospitalizations, which contains the number of hospitalizations each patient has within the MIMIC-IV database prior to their current hospitalization. Table 4.2 below displays a sample of the final set of variables extracted from the structured data that were used for modeling.

Table 4.2. The final set of variables used for modeling extracted from the structured data

Admission Type	Admission Location	Discharge Location	Insurance	Race	Length of Stay	Age	Gender	ICU Flag	Previous Hospitalizations	Death Flag
URGENT	TRANSFER FROM HOSPITAL	HOME	Medicaid	WHITE	0.786111	52	F	0	0	1
EW EMER.	EMERGENCY ROOM	HOME	Medicaid	WHITE	1.015278	52	F	0	1	1
EW EMER.	EMERGENCY ROOM	HOME	Medicaid	WHITE	2.222222	52	F	1	2	1
EW EMER.	EMERGENCY ROOM	HOSPICE	Medicaid	WHITE	1.754167	52	F	0	3	1
EW EMER.	WALK-IN / SELF REFERRAL	HOME HEALTH CARE	Medicare	WHITE	4.538889	72	M	0	0	1

In total, a set of eleven variables were selected from the structured data, including the dependent variable. This set includes the newly created variables, demographic information from the *patients* table, and some of the variables originally in the *admissions* table. When comparing this set to Table 4.1, it is clear that the two ID variables, the admission and discharge times, and the hospital expire flag variable (which defines which patients died during their hospital visit) were removed as I did not consider them meaningful predictors.

Having established a set of variables to use for the structured data portion of my modeling, I completed the final steps of data preparation by removing certain entries. First, I removed all patients who died during their hospital stay as their inclusion would not be appropriate for a model that predicts one-year mortality post hospital discharge. Following this, I removed all patients with any missing data values for the selected variables. Finally, I removed all patients without a free-text discharge summary using the *discharge* table of the MIMIC-IV-Note dataset. This step was critical to ensure a one-to-one comparison between the models created using just the structured data and the combined structured and unstructured data. Figure 4.2 below summarizes how the dataset was narrowed down.

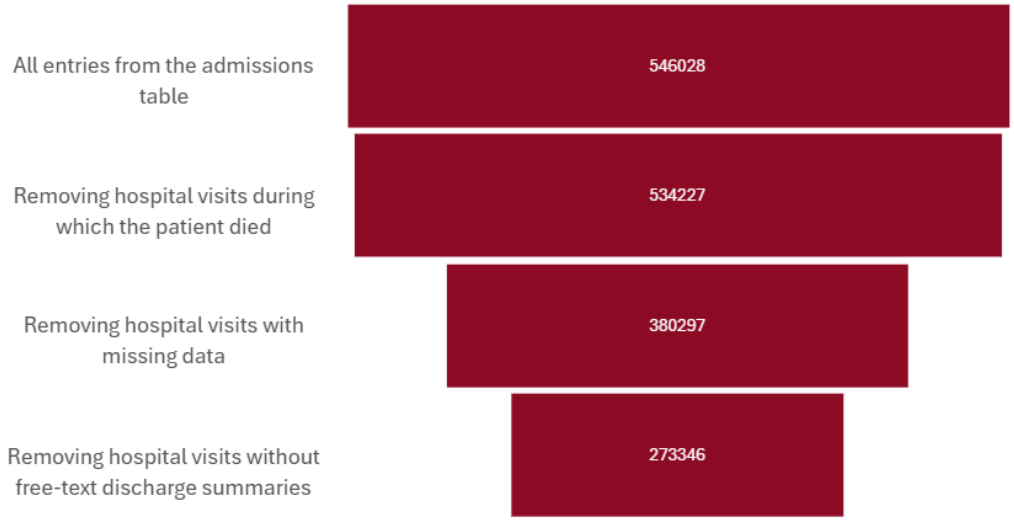


Figure 4.2. *Funnel plot summarizing the data preparation process.*

As shown in Figure 4.2, the data preparation process resulted in the removal of roughly half of all hospitalizations in the MIMIC-IV dataset. Nonetheless, the resulting set of patients was still more than adequate for the modeling task of this study.

Using the modeling dataset of patients defined above, I then carried out some exploratory data analysis to better understand the data prior to modeling. Figure 4.3 below displays the different locations patients were admitted from and their frequencies.

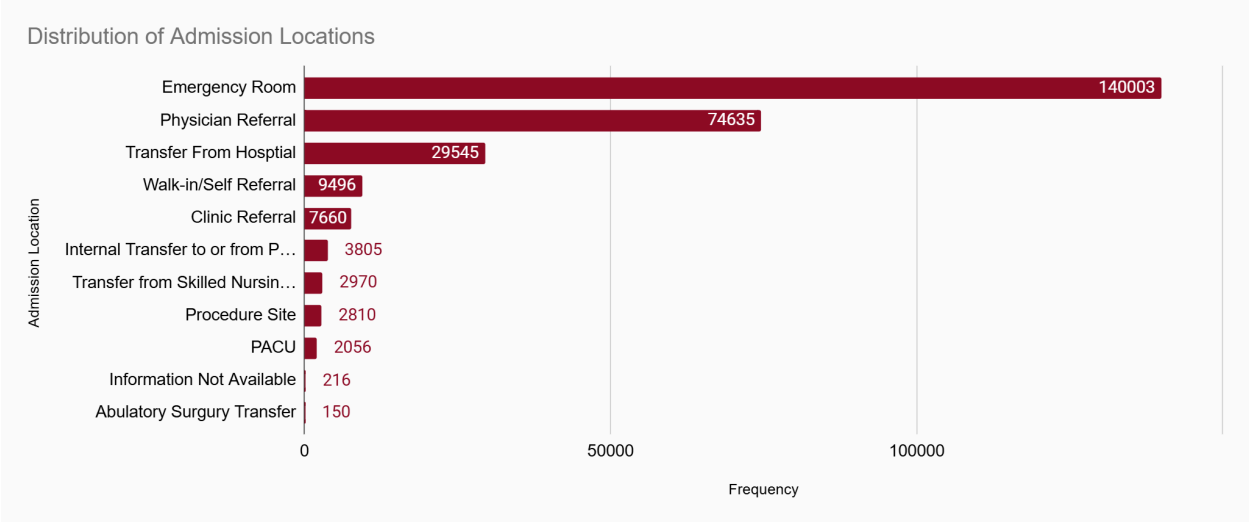


Figure 4.3. *The distribution of admission locations in the modeling dataset.*

This bar plot shows that the most common admission location of all patients in the modeling set is the emergency room. This is not very surprising considering that hundreds of millions of individuals go to the emergency room each year. The next most common admission location is physical referral which means that the patient was deemed to require hospitalization by their physician and thus referred to the hospital. All other admission locations are far less common. I also conducted a similar analysis of discharge location. This can be found in Figure 4.4 below.

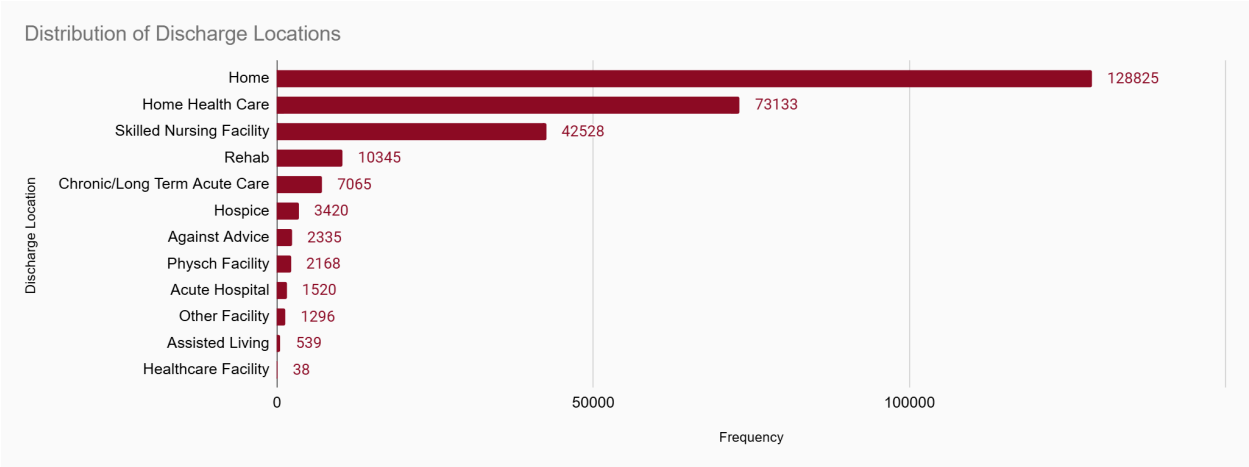


Figure 4.4. *The distribution of discharge locations in the modeling dataset.*

Based on Figures 4.3 and 4.4, it seems that both admission and discharge location variables follow a similar distribution. In the case of the discharge location, the most common location is home. This is logical as most people that go to the hospital receive care to the point where they no longer need it and can thus go home. Interestingly, the second most common location is home health care which means that patients are leaving the hospital and receiving more care at home by a professional.

Following the analysis of admission and discharge location, I examined demographic features, like insurance, race, gender, and age. In this case, I am considering insurance a demographic feature due to its relationship with socioeconomic status. The distribution of insurance types can be found in Figure 4.5 below.

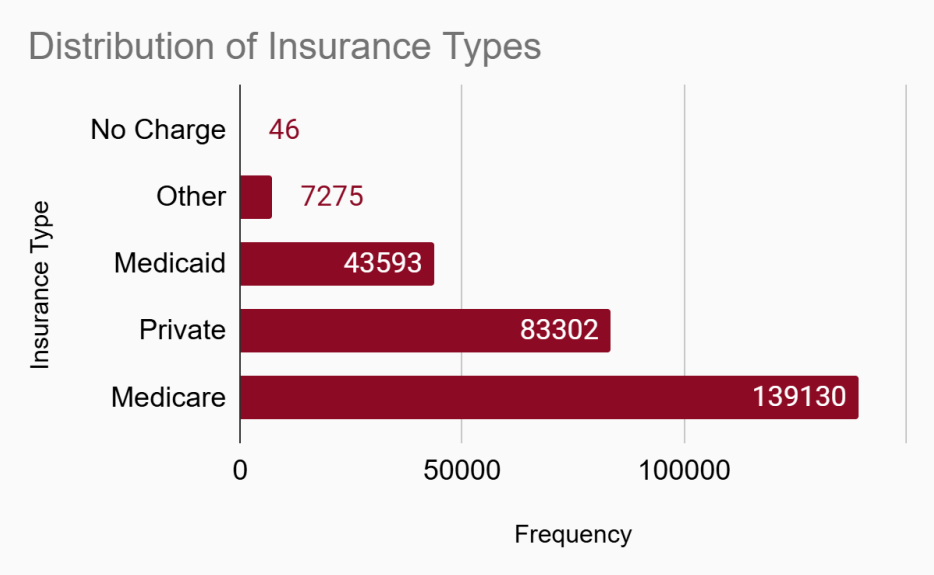


Figure 4.5. *The distribution of insurance types in the modeling dataset.*

As shown in the above bar plot, Medicare is the most common type of insurance of the patients within the modeling dataset. This is logical given that Medicare is specifically for people over the age of sixty-five, a population more likely to experience illness and require hospitalization. After Medicare, the next most common insurance is private, followed by Medicaid. All other admissions then fall under alternate forms of insurance or no charge. Based on the distribution of insurances, it does not seem that the socioeconomic status of the population within the modeling dataset is outside the norm of the United States. The same can be said of the overall distribution of races and genders within the modeling set. Of all patients used for modeling, roughly 66% are white and 12% are black. Additionally, roughly 50.5% are female and 49.5% are male. This is fairly aligned with the general population of the US. The final demographic feature analyzed was age. Figure 4.6 below displays the distribution of patient age within the modeling dataset.

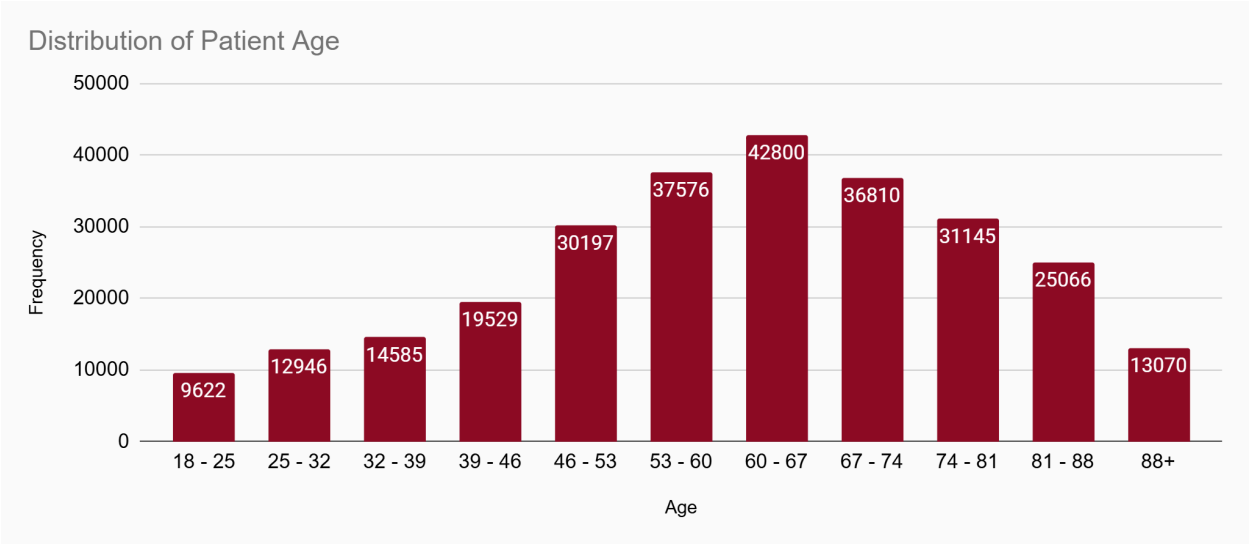


Figure 4.6. *The distribution of age in the modeling dataset.*

As shown in the above histogram, the distribution of ages is left skewed and roughly centered around the age of sixty-three. This means that there is a disproportionately large number of patients above the age of roughly sixty-three which, again, aligns with the fact that older individuals are more likely to require hospitalization. This is also in alignment with the fact that the most common form of insurance in the modeling set is Medicare.

At this point, I created several models to predict one-year mortality based on the structured data to form an idea of the baseline predictive power of the structured data and to understand which model types performed best. Prior to fitting the models, I used one-hot encoding to convert all categorical variables into numeric variables and scaled each numeric feature so that it had a mean of zero and standard deviation of one. This is standard practice when modeling with categorical data and numeric data that varies in scale. Also in accordance with modeling best practices, I set up a cross-validation framework with five folds to better assess model performance and prevent overfitting. Using the five fold cross validation, I fit a logistic regression model, a decision tree, a random forest, a gradient boosting classifier, and a

neural network. All of these models are common practice for classification tasks like the one in this study.

To evaluate each model, I mainly considered the average recall of all five folds, given the nature of the models created. Because the models were predicting whether a patient would die within a year of being discharged, I wanted to ensure the models predicted as many of those patients who would go on to die correctly. I considered this as the most important predictive outcome with the underlying logic being that if a model predicting one year mortality were used in practice, identifying high risk patients could allow them to receive more urgent care and thus potentially save lives. There are other ethical factors to consider here, but this was the main criteria I maintained throughout the entire modeling process. Of course, when measuring recall, it is also important to consider precision or F1 score to ensure the overall model performance improves, not just that of the positive class. As a result, F1 score will also be discussed when evaluating model performance. In addition to recall and F1 score, accuracy, precision, and AUROC were also calculated for each trained model.

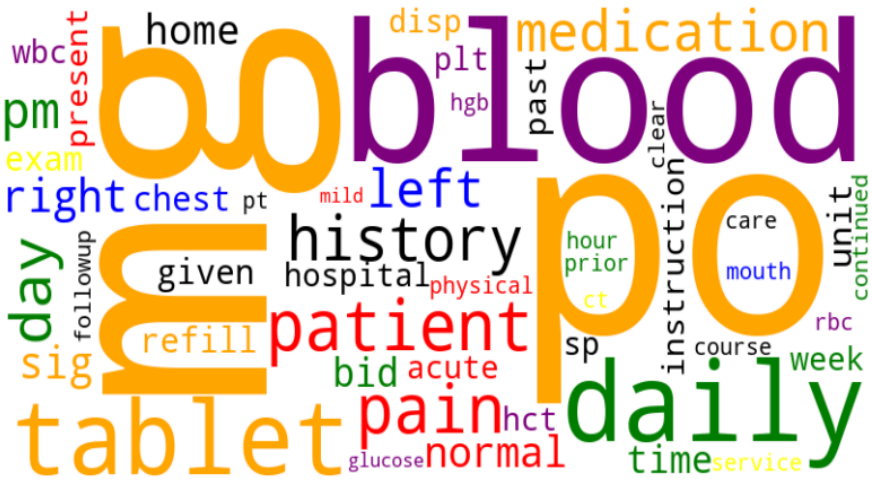
Based on the results of the baseline models (which are provided in the Analysis & Discussion section), the need for data resampling was evident. Upon reviewing the data, I found that roughly 85% of all hospital admissions resulted in the patient surviving beyond one year post discharge, while the remaining 15% resulted in mortality within the year. Given the class imbalance, I tested multiple resampling methods to analyze their impact on model performance while determining which resampling method performed best. I used the neural network when testing the resampling methods as it performed best of all baseline models. I first tested two oversampling techniques, random oversampling and Synthetic Minority Oversampling Technique (SMOTE). I then tested random undersampling and edited nearest neighbors. Finally,

I tested two hybrid approaches by combining random oversampling with random undersampling and SMOTE with random undersampling. For all resampling methods applied (other than edited nearest neighbors) the minority and majority classes were made to be equal in size. Based on the model results, I ultimately decided upon using the hybrid random oversampling and random undersampling technique when creating all remaining models in this study.

After testing the resampling methods, I recreated each baseline model using the hybrid random oversampling and undersampling technique for comparison purposes. I then began incorporating the unstructured clinical notes into each model type, leveraging different natural language processing techniques. Prior to applying specific techniques, I first cleaned and normalized the text. This process involved first changing all characters to lowercase and removing all special characters. After this, each word in each note was separated into tokens, all stop words (words such as 'the', 'is', 'and', etc.) were removed, and each word was lemmatized. The lemmatization process was important for this task as it ensures that each word is in its root form by changing all verbs to the same tense, all plural nouns to singular, and more. This allows NLP methods to derive clearer insights from each clinical note by reducing variations of the same words. The final step of the cleaning and normalization process was to recombine each token into a single string as they were before.

With the discharge summaries cleaned and normalized, the first NLP method I applied was bag of words. Due to the large amount of clinical notes used for this analysis (which in total contain thousands of unique words), I decided to only consider the fifty most frequent words to significantly reduce the number of features for modeling while capturing the most important themes from the clinical notes. When obtaining the initial output, I found that even after data cleaning, some of the most common words in the notes were still not very meaningful. For

example, words like ‘date’ were included in the initial bag of words as all discharge summaries discuss the date on which it was written. Furthermore, there were many instances of numbers being very common. I subsequently removed these words and numbers and recreated the bag of words. Figure 4.7 below displays the fifty most common terms found using bag of words.



Legend	
Category	**Color**
Diagnosis & Clinical Assessment	Red
Medications & Prescriptions	Orange
Procedures & Imaging	Yellow
Timing & Frequency	Green
Anatomy & Body Systems	Blue
Laboratory Tests & Results	Purple
Other	Black

Figure 4.7. *A word cloud containing the fifty most common terms in the discharge summaries.*

In the word cloud above, size denotes the frequency of each term and color represents the category they fall within according to the given legend. The shown categories were created by myself based on the resulting words to drive insights and are separate from the natural language processing method of bag of words. The diagnosis and clinical assessment category contains words related to any diagnoses about a patient’s condition. As shown in red, the most common words within this category are patient and pain. The medications and prescriptions category contains words related to the medications a patient is taking. As shown in orange, the most common terms here are mg (milligram) and po (this term means a medication is to be taken by mouth). The procedures and imaging category contains words related to any clinical tests and procedures conducted to a patient. The only two words that fall under this category are exam and service, shown in yellow. Given the small size of these words, it is evident that they are relatively

uncommon compared to some of the other words shown. The timing and frequency category contains words indicating timeframes for monitoring or treatment. The most common word in this category is daily, as shown in green. The anatomy and body systems category contains terms referring to body parts or other physiological aspects. As shown in blue, the most common words in this category are left and right, perhaps indicating what side of the body a patient is experiencing a medical issue. The laboratory tests and results category contains words related to lab tests. Shown in purple, the most common word here is evidently blood, likely indicating that blood tests are very common. The final category is other and simply contains words that do not fall within the other listed categories. Words like home and history are within this category and are colored black. A subset of the table representation of the output of bag of words can be found in Figure A1 of the Appendix.

After combining the bag of words output with the structured data previously used for modeling, I reran all five model types using the same data preparation and cross validation process. As mentioned prior, the hybrid random oversampling and undersampling was also applied to improve model performance. The same process was also applied for three more natural language processing techniques. Of these three, the first was binary bag of words. For this method, I also only selected the top fifty words but the selection criteria changed with the use of binary. Instead of selecting the most common words by using their total frequency in all documents, this method determined the most common words by how many notes they were present in. In other words, the sum of the binary values of each word was used. Figure 4.8 below displays the most common terms found using the binary bag of words method.

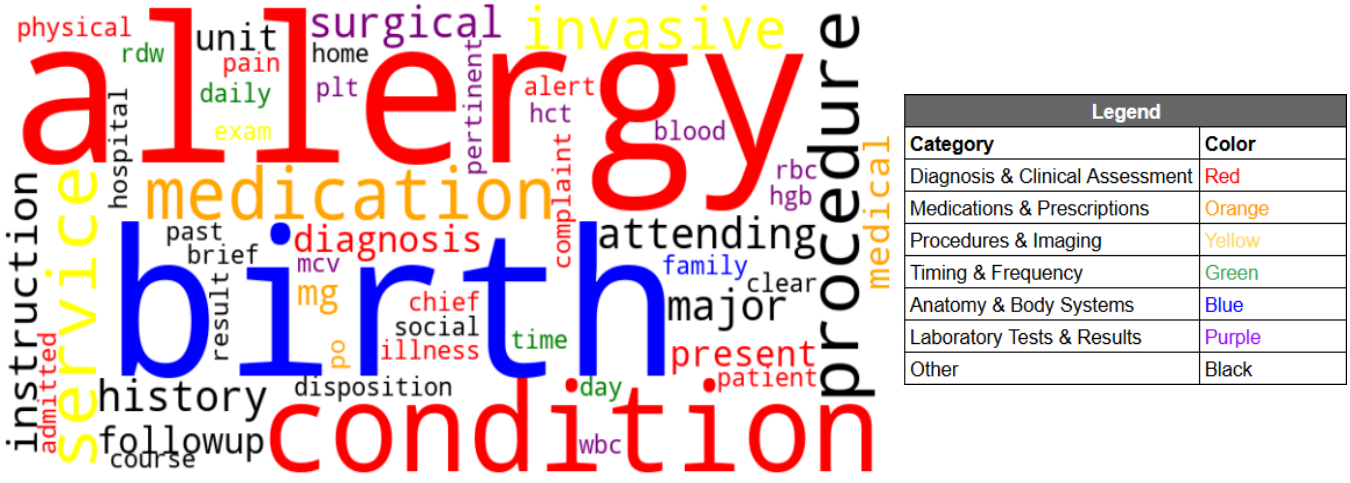


Figure 4.8. *A word cloud containing the fifty most common terms using binary bag of words.*

Evidently, the binary bag of words method resulted in a significantly different set of words compared to the standard bag of words method. As shown by their size, the most common words in this case are allergy and birth. Interestingly, these two words were not even within the set of fifty words found using the standard bag of words method. This is due to one of the main limitations of using the binary version of bag of words for the discharge summaries in the MIMIC-IV-Note dataset. As shown in the example discharge summary in Figure 4.1, each clinical note begins with a standard set of information, including a patient's name, admission date, date of birth, etc. Because of this, the words "name," "admission," "date," "birth," and more will be valued very highly using binary bag of words as they are present in all notes. While I removed most of these words from consideration when modeling, some instances remained, like birth and allergy. I decided against removing words like these as they could be important in other medical contexts. For example, the word "birth" could simply be used to denote one's date of birth, but could also be related to the birth of a child. Similarly, the word "allergies" (after lemmatizing, "allergies" turns into "allergy") is listed in each discharge summary to denote any

medicine allergies a patient has but could also be used to discuss an allergic reaction a patient had during their hospital stay. A table representation of a subset of the output of binary bag of words can be found in Figure A2 of the Appendix.

After testing the bag of words methods, I then tested the use of term frequency-inverse document frequency (TF-IDF). Like the bag of words methods described above, I also only considered the fifty most common words for this method. To determine which words were most common, I simply took the sum of the TF-IDF value of each word in each document and chose the words with the fifty highest values. Figure 4.9 below displays the most common words found using the TF-IDF method.

Figure 4.9. *A word cloud containing the fifty most common terms found using TF-IDF.*

Evidently, the most common words found using TF-IDF are related to medications and prescriptions as tablet refers to a pill and sig refers to the label of a prescription. Interestingly, one of the most common terms is also pm, which likely refers to post meridiem, indicating a time that occurred in the afternoon or evening. This could also be an abbreviation for past medical history. A table representation of a subset of the output of TF-IDF can be found in Figure A3 of the Appendix.

The final natural language processing technique applied was sentiment analysis, performed using the sentiment module of the NLTK Python library. Unlike the previous three methods described, sentiment analysis does not create features based on word frequency, but rather creates one feature that judges the overall sentiment of each discharge summary. Figure 4.10 below displays the distribution of sentiments across all discharge summaries in the MIMC-IV-Note dataset.

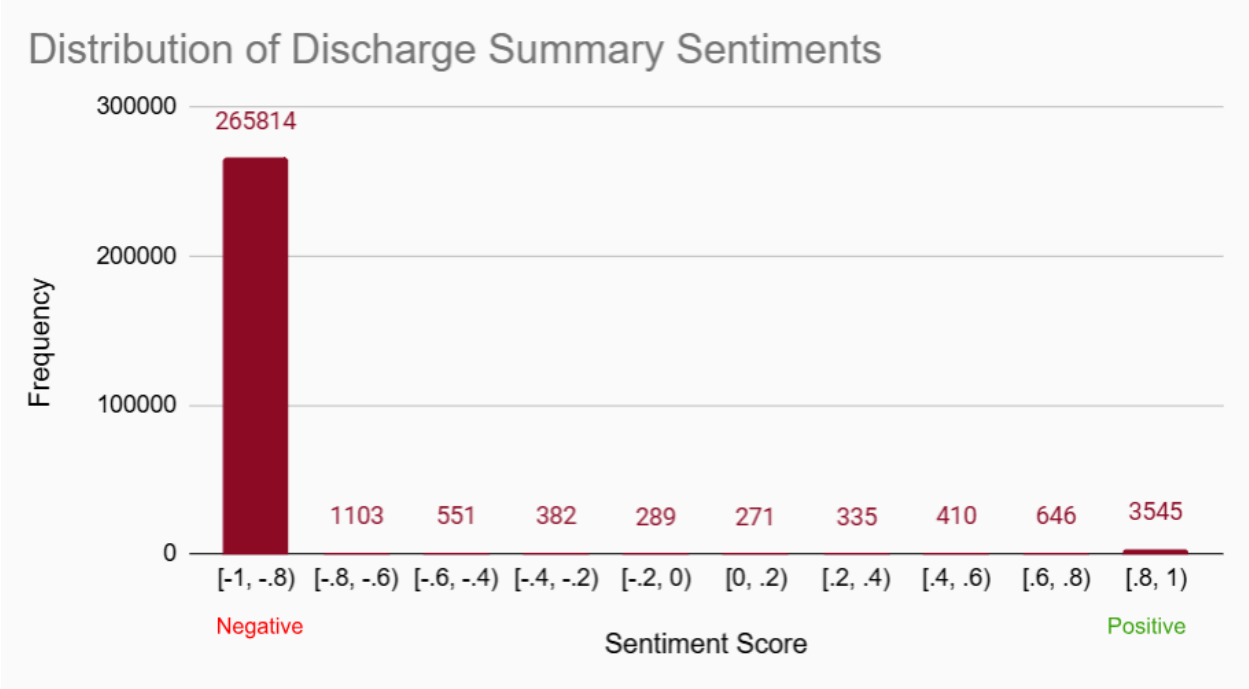


Figure 4.10. *The distribution of sentiments across all discharge summaries*

Evidently, the distribution of sentiments is extremely right skewed as the vast majority of clinical notes have a sentiment score near -1, the most negative it can be. This is likely due to the unique qualities of medical data. First, medical dialogue tends to contain words that are innately associated with negative outcomes. Because the sentiment analysis is carried out based on the words in each note, it is logical that most would be negative. Additionally, sentiment analysis tools are not trained using medical data and thus do not fully understand the nuanced expressions

related to patient outcomes. Despite these challenges, sentiment analysis remained a valuable component to consider when modeling. The sentiment analysis model was the last to be fit, marking the completion of the modeling process. With all models created, the final step of this methodology was to carry out a feature importance analysis to better understand which features had the most impact on the prediction of one year mortality.

V. Analysis and Discussion

This study aimed to evaluate the impact of using free-text clinical notes on the predictive modeling of healthcare outcomes. In doing so, five different model types - logistic regression, decision tree, random forest, gradient boosting classifier, and neural network - were applied to create one-year mortality predictive models using both structured data and the combination of structured data and free-text clinical notes. Additionally, this study leveraged and analyzed four natural language processing techniques and six resampling methods in order to maximize model performance. Below, I present the key findings derived from the previously described techniques using the MIMIC-IV and MIMIC-IV-Note datasets. The results are structured as follows: first, an overview of the baseline model performances without resampling, next, an analysis of the resampling methods and their application to the baseline models, and finally, an examination of the effects of the free-text clinical notes on model performance.

As previously described, this study began with the creation of models trained using only structured data so that an idea of baseline model performance could be formed and thus used for comparison with later models. Table 5.1 below contains the results of these baseline models, created using the five different modeling techniques, and evaluated using accuracy, precision, recall, F1 score, and AUROC.

Table 5.1. Results of baseline models, trained using only structured data and no resampling

	Logistic Regression	Decision Tree	Random Forest	Gradient Boost	Neural Network
Accuracy	0.856	0.856	0.858	0.858	0.856
Precision	0.708	0.675	0.79	0.739	0.595
Recall	0.099	0.107	0.089	0.106	**0.169**
F1 Score	0.173	0.185	0.16	0.185	0.262
AUROC	0.546	0.549	0.542	0.55	0.574

As shown in the above table, all baseline models perform very poorly. The only metric which seems to be positive is accuracy as every model type achieved an accuracy of roughly .86. However, when putting this value into the context that roughly 85% of the modeling dataset are patients who survived one year post-discharge, it is evident that the models are simply predicting the vast majority of patients to survive, thus incorrectly predicting the majority of patients who actually died. This is reflected in the very low recall scores of all the models. As explained prior, recall is the primary evaluation metric used in this study given the context of the models. Using the neural network as an example, its recall score of .169 means that only 16.9% of all patients who died are being predicted to do so. This means that if this model were to be put to practice, only 16.9% of all high risk patients would be flagged and thus given the additional care they require. Consequently, the remaining 83.1% would not receive any additional care.

The high accuracy and low recall scores of the baseline models shown in Table 5.1 exemplify the need for resampling. As a result, I tested six different resampling techniques using the same dataset. Because the neural network performed best of all models based on recall score, I used the neural network to test all resampling methods. Table 5.2 below shows the results of each resampling technique tested.

Table 5.2. The results of each resampling technique applied to the neural network, trained using only structured data

	Baseline	Random Oversampling	SMOTE	Random Undersampling	Edited Nearest Neighbors	Random Oversampling and Random Undersampling	SMOTE and Random Undersampling
Accuracy	0.856	0.71	0.726	0.696	0.805	0.702	0.72
Precision	0.595	0.304	0.31	0.296	0.39	0.299	0.308
Recall	0.169	0.701	0.646	**0.716**	0.487	**0.714**	0.664
F1	0.262	0.424	0.418	0.419	0.432	0.422	0.42
AUROC	0.574	0.706	0.693	0.705	0.676	0.706	0.697

Using recall as the primary evaluation metric, it is evident that the use of resampling methods greatly improved model performance. In fact, the use of all resampling methods caused recall to increase by roughly forty-nine percentage points, on average. In this case, using random undersampling resulted in the model with the greatest recall score. Nonetheless, I ultimately decided against using random undersampling and instead chose the hybrid random oversampling and random undersampling technique for all other models in this study. I made this decision because random undersampling alone resulted in the loss of roughly 70% of the data in the modeling set. Meanwhile, the hybrid random oversampling and random undersampling technique resulted in a very similar recall score while maintaining much more of the data. Additionally, this hybrid technique outperformed the random undersampling technique in all other evaluation metrics calculated. It should be noted that the introduction of resampling techniques caused a notable decrease in accuracy and precision. For accuracy, one must remember that the resampling techniques removed the class imbalance. With this in mind, the new accuracy scores are actually quite positive as it is clear that the models are no longer predicting the same outcome for the majority of patients. For precision, the lower scores suggest

that more patients are being incorrectly predicted to die within a year-post discharge. Unfortunately, this is a necessary tradeoff if the desired goal is to predict as many of the patients who actually will die correctly. Considering the significant increase in recall, the subsequent decrease in precision is both understandable and acceptable.

Having chosen to use the hybrid random undersampling and random oversampling technique based on the results shown in Table 5.2, I first recreated each baseline model from Table 5.1 so that they could be used as a basis of comparison for later models. The recall scores of each model type with this resampling technique applied can be found in Figure 5.1 below. The results in table form with all model evaluation metrics calculated can be found in Table A4 of the Appendix.

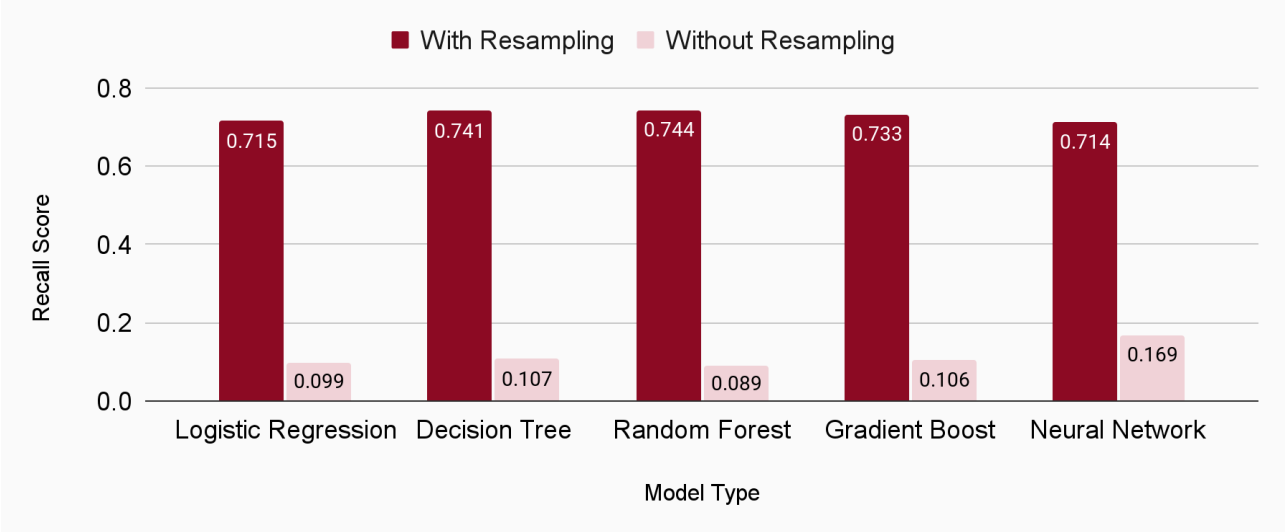


Figure 5.1. *Recall scores of baseline models with resampling (using hybrid random oversampling and random undersampling), trained using only structured data.*

As shown with the neural network in Table 5.2, the use of random oversampling and random undersampling greatly improved model performance of all model types. On average, the recall

score of each model increased by roughly 61.5 percentage points. However, while the neural network model had the highest recall score prior to resampling, it had the lowest recall score after implementing random oversampling and random undersampling. This suggests that the neural network may be memorizing the resampled data rather than learning the underlying patterns of the data. Interestingly, the model with the highest recall score after applying random oversampling and random undersampling was the random forest model, which had the lowest recall score prior to resampling. This model's recall score rose by roughly 65.5 percentage points, more than any other model.

I then began incorporating the discharge summaries into the modeling process. In doing so, I tested four different natural language processing techniques: bag of words, binary bag of words, TF-IDF, and sentiment analysis. The recall scores of the models created using these techniques can be found in Figure 5.2 below. The results in table format with all model evaluation metrics can be found in Figure A5 in the Appendix. Note that these models were trained using both the structured data used in the baseline models and the insights derived from the discharge summaries using each individual natural language processing technique.

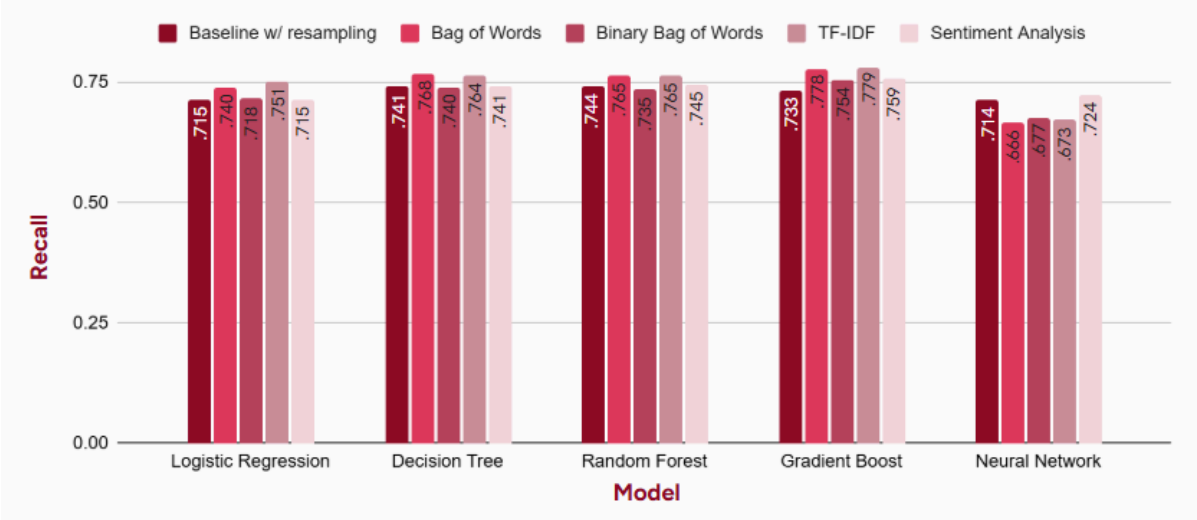


Figure 5.2. *The recall scores of the models trained using both the structured data and discharge summaries, processed using four different NLP techniques.*

As shown in the above table, the best performing models were the gradient boosting classifiers trained using TF-IDF and bag of words, resulting in a recall score of .779 and .778, respectively. Interestingly, for all natural language processing techniques used, the gradient boosting classifier performed best out of all other model types. Notably, this is the only model type which improved with the introduction of each natural language processing technique compared to its baseline performance with resampling. The logistic regression improved for all NLP techniques other than sentiment analysis, where performance remained constant. The decision tree improved for bag of words and TF-IDF, while the random forest improved for all methods except binary bag of words. Interestingly, the neural network only improved with sentiment analysis. Given the limitations of binary bag of words and sentiment analysis that were discussed prior, it is logical that their inclusion resulted in worse performance relative to bag of words and TF-IDF. The reason the neural network only improved with sentiment analysis is likely related to the difference in dimensions of the training data: sentiment analysis resulted in only one new feature

for each data entry while the other methods resulted in fifty new features, based on the fifty most common words. The increased dimensionality likely caused the neural network to be unable to identify the underlying patterns of the data, thus causing it to not generalize well, decreasing model performance.

Overall, model performance generally improved with the use of the clinical notes. This is shown in Figure 5.3 below, which displays the improvement in recall score caused by the inclusion of the discharge summaries for the best performing model of each model type. All recall score improvements are relative to each model's baseline recall score with resampling.

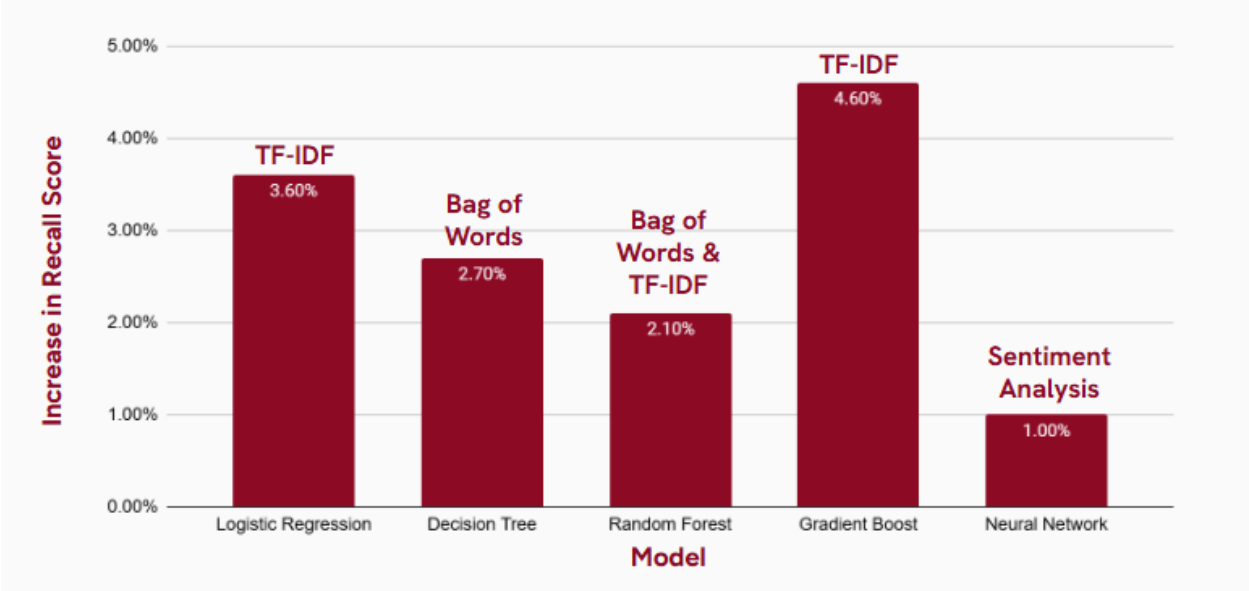


Figure 5.3. *Highest improvement in recall score of each model type with the introduction of discharge summaries.*

As shown, the best performing neural network (with sentiment analysis) resulted in a recall score one percentage point higher than its baseline with resampling. Logistic regression improved by 3.6 percentage points with TF-IDF, the decision tree improved by 2.7 percentage points with bag

of words, and the random forest improved by 2.1 percentage points with both TF-IDF and bag of words (these two models had the same recall score). Finally, the highest performing gradient boosting model, trained using TF-IDF, resulted in a recall score 4.6 percentage points higher than its baseline model with resampling. Overall, the use of the free-text clinical notes resulted in improved recall score for all model types.

Of course, it is also important to consider precision or F1 score to ensure each model is not making a disproportionate amount of false negatives in order to increase the amount of true positives it predicts. Figure 5.4 below displays the change in F1 score for each model shown in Figure 5.3. All F1 score improvements are relative to each model's baseline F1 score with resampling.

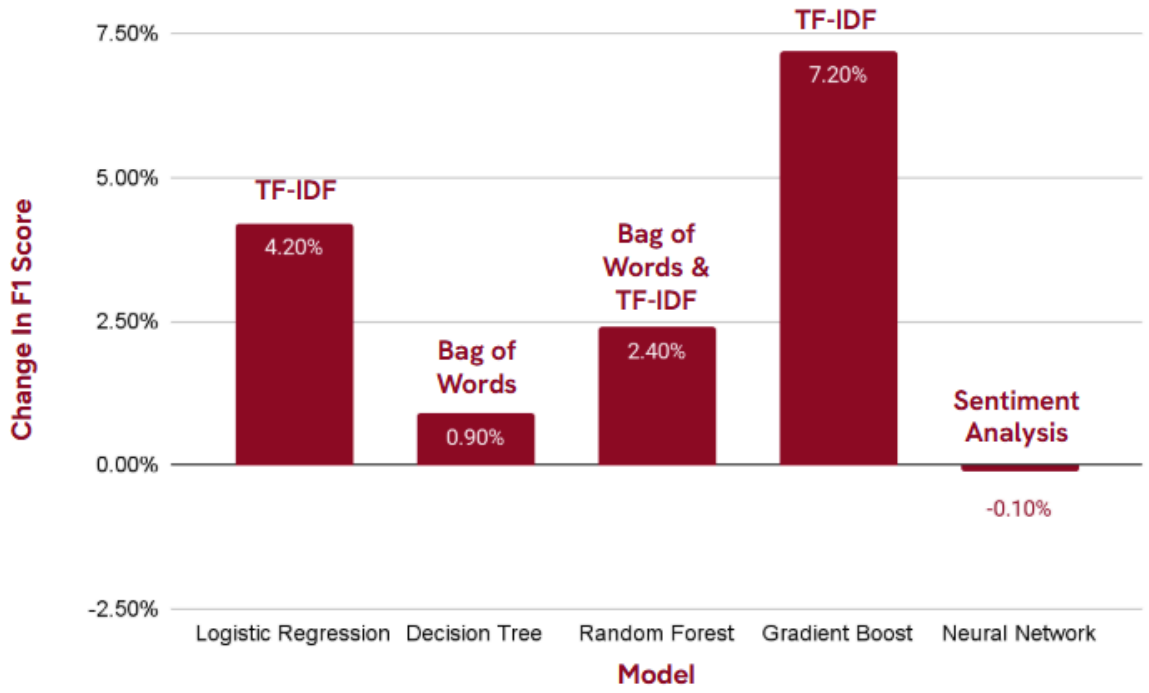


Figure 5.4. *Change in F1 score for the model of each type that showed the greatest improvement in recall with the inclusion of discharge summaries.*

As shown in Figure 5.4, the majority of the models which improved the most in recall score also improved in F1 score, indicating that the overall predictive performance of these models improved while increasing the number of true positives predicted. Interestingly, the model which improved most in F1 score was also the model which improved most in recall score, the gradient boosting classifier with TF-IDF. In addition to having the highest recall score, this model also resulted in the highest F1 score of all models created in this study. The only model shown in Figure 5.4 that did not show an improvement in F1 score was the neural network with sentiment analysis, whose F1 score was just 0.1 percentage points lower than that of its baseline model with resampling. Nonetheless, the neural network did improve in F1 score with the introduction of the discharge summaries when using NLP techniques other than sentiment analysis. Overall, the use of the discharge summaries resulted in improved predictive performance for all model types.

The final step of this study was to conduct a feature importance analysis to better understand which variables had the most effect on one-year mortality. Because the gradient boosting model trained using TF-IDF performed best out of all models, it was used for the feature importance analysis. Figure 5.5 below displays the feature importance of the ten most important features from the gradient boosting model using TF-IDF. Recall this model was trained using both the structured data and the words extracted from the discharge summaries using TF-IDF, so the most important features arise from both data sources. Also note that all of the categorical features from the structured data were one-hot encoded prior to modeling.

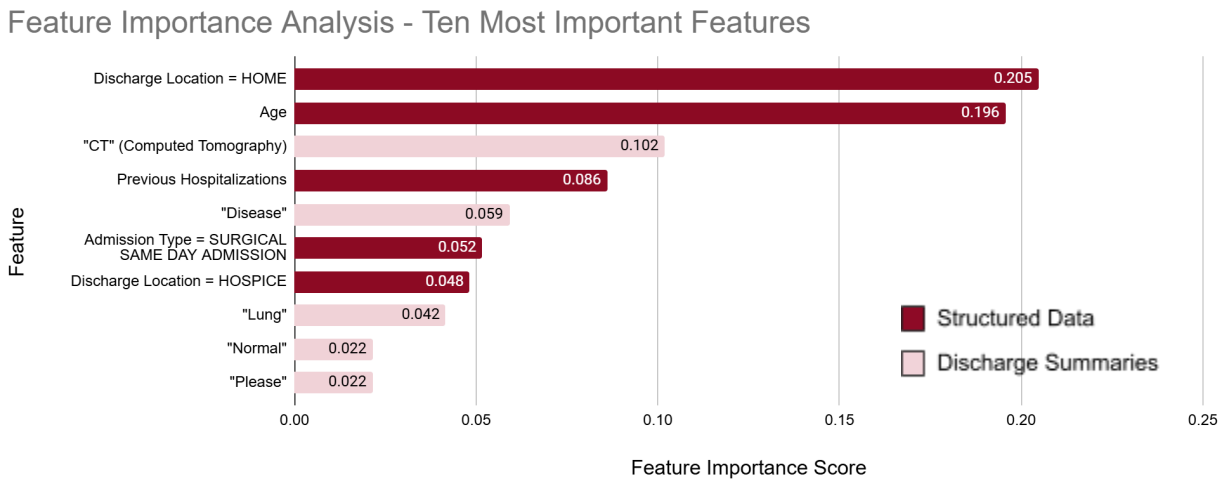


Figure 5.5. *Feature importance of the ten most important features from the gradient boosting model using TF-IDF.*

In the above bar chart, dark red bars represent variables from the structured data and the pink bars denote words from the free-text discharge summaries. As shown, out of the top ten most important features, five are from the structured data and five are from the discharge summaries. However, the features from the structured data are overall more important to the predictions of the model as their average feature importance score is higher.

Evidently, the feature deemed most important to the model was discharge location being equal to home, denoting that a patient was sent home after discharge from the hospital. It is likely that this feature decreases the probability of one year mortality as patients are only discharged to their home if deemed healthy enough to no longer require consistent care. The second most important feature was patient age, obtaining a feature importance score very similar to discharge location home. As patient age increases, it is likely that the chance of one-year mortality increases as well given that health tends to deteriorate as people age. The third most important feature was the term “ct” from the discharge summaries, with a feature importance score that

indicates it is roughly half as important as the discharge location home and patient age features. The high TF-IDF score of this term in a discharge summary likely indicates that multiple ct scans were performed or that the results of a ct scan were significant. This suggests that the patient's condition is not well, thus suggesting that this feature likely increases the probability of one-year mortality. The next most important feature was the previous_hospitalizations variable which contains the number of hospitalizations a patient had prior to their current one within the MIMIC-IV dataset. Logically, the higher the number of previous hospitalizations, the more likely a patient is to die within a year of discharge as multiple hospitalizations suggests a patient has a persistent health issue severe enough to require hospitalization. The fifth most important feature was the word "disease" from the free-text discharge summaries. The high inclusion of this word in one's discharge summary likely suggests a more complex patient situation as the patient could be suffering from multiple diseases or have a family history with many diseases. As a result, the probability of one year mortality likely increases as the TF-IDF of this term increases.

Following the term "disease", the next five most important features are admission type surgical same day admission, discharge location hospice, "lung," "normal," and "please." The first two of these features are from the structured data, with the first denoting that a patient was admitted to the hospital for a same day surgery, and the second denoting that a patient was discharged to a hospice. Logically, both of these occurrences increase the chance of one-year mortality. The last three features are all from the free-text discharge summaries. The word "lung" suggests a patient has issues with their lungs, likely increasing their probability of one year mortality. The word "normal" suggests that a patient's condition is normal, suggesting they are less likely to die within a year of discharge. The final word, please, does not have such a logical relationship with one year mortality. This word is mostly used in the discharge summaries when

discussing the discharge instructions given to patients or when making suggestions to other doctors for future visits. It is overall unclear if this would suggest a patient is more or less likely to die within a year of discharge.

VI. Conclusion

Summary of Findings

Predictive modeling in healthcare will only become more prominent in the coming years. Because of this, it is critical to fully understand the value of all healthcare data sources that can be used to train models. This study examined the effect of using free-text clinical notes to train predictive models in healthcare, leveraging over 300,000 discharge summaries from the MIMIC-IV-Note dataset. Through the application of these discharge summaries to five different modeling techniques and four different natural language processing methods, this study found their inclusion to incrementally improve model performance. All model types trained - logistic regression, decision tree, random forest, gradient boosting classifier, and neural network - improved with the use of at least one of the NLP techniques tested. Overall, the model that performed best was the gradient boosting classifier trained using TF-IDF, obtaining a recall score of .779 - 4.6 percentage points higher than that of its baseline with resampling. Applied to the dataset used to train the models, this increase in recall score equates to roughly 1,900 more admissions being correctly identified as high risk for one-year mortality. Not only did the gradient boosting classifier achieve the best recall score when trained using TF-IDF, it was also the only model type that saw improved predictive power with all NLP methods tested. Out of all these NLP techniques, both bag of words and TF-IDF seemed to be the most effective at deriving meaningful insights from the discharge summaries as the models trained using the outputs of these methods resulted in the highest average improvement.

This study also highlighted the importance of using resampling techniques when training predictive models in healthcare. Like many healthcare outcomes, one-year mortality is heavily

unbalanced as the vast majority of patients tend to survive more than one-year post discharge. This imbalance resulted in the extremely poor performance of the initial baseline models. After testing and applying many different resampling techniques, this study found the hybrid random oversampling and random undersampling to result in the best model performance. Applying the technique caused an average increase in recall score of 61.54%. Overall, the model improvements caused by the application of resampling techniques and the inclusion of free-text clinical notes show that they are absolutely critical for models in healthcare. With the improved performance of these models, proper care can be provided to all patients who need it most, thus leading to improved health outcomes and even saving lives.

Limitations

The main limitations of this work lie in the techniques applied for resampling and natural language processing. Due to the large size of the data used to train the models, relatively simple resampling and NLP methods were used. More advanced techniques were applied but either resulted in prohibitively long runtimes or required more memory than my computational resources could provide. For example, when analyzing the resampling techniques, I tested random oversampling, random undersampling, SMOTE, edited nearest neighbors, and two hybrid techniques created by combining two of the previously mentioned methods. Random oversampling and random undersampling are two of the simplest resampling techniques as data entries are randomly selected and duplicated or removed. SMOTE and edited nearest neighbors are more complex, in comparison, as SMOTE generates synthetic data entries and edited nearest neighbors applies a clustering algorithm. In addition to these methods, I also tested Tomek Links, cluster centroids, and adaptive synthetic sampling (ADASYN), but all of these techniques proved

to be too complex in that they resulted in challenging runtimes. It is possible that these techniques would have provided better model performance than those shown in this study.

The natural language processing techniques shared similar challenges. Firstly, as discussed prior, sentiment analysis and binary bag of words contain their own unique constraints, potentially leading to worse model performance. Nonetheless, both of these techniques in addition to bag of word and TF-IDF are all relatively simple and some even share similar underlying mechanisms. Besides these techniques, I also tried using Clinical BERT, a language representation model trained specifically for medical text, but unfortunately ran into memory constraints given the number of clinical notes being processed. This method, given that it was trained specifically for the medical domain, likely would have resulted in higher model performance than the NLP methods applied in this study.

Future Work

This thesis lays the foundation for further analysis of the effects of free-text clinical notes in healthcare modeling, presenting many potential areas for future research. For example, this study can be built upon by testing more complex resampling and NLP techniques, as discussed in the previous section. To do this, more powerful computation resources than those used for this study would be required if using the same training dataset. Another option that would likely allow for more complex techniques to be applied would be to use a smaller dataset to train the models. The simplest way to reduce the size of the dataset is to simply take a random subset. However, one can also more deliberately select a smaller dataset with the goal of analyzing a specific population of patients. For example, instead of using the entire patient population (after applying data cleaning) like in this study, future work could focus on patients with specific demographic features or specific diseases. This would greatly reduce the size of the data used for

modeling while still allowing for conclusions to be made regarding the use of clinical notes in healthcare models. Additionally, this type of study could also reveal interesting trends about the patient population used for modeling, thus adding another layer to this work.

Another area for further research is to create models that predict outcomes other than mortality. These outcomes include, readmittance, length of stay, ICU admission, disease, and more. Of course, some of these outcomes would require clinical notes other than discharge summaries as outcomes like length of stay or ICU admission would already be known if the patient has been discharged. Not only would the prediction of different outcomes potentially provide different insights regarding the use of clinical notes, but the use of clinical notes other than discharge summaries may do the same. To add another layer of complexity to the outcomes being predicted, time can also be considered. For example, instead of predicting whether a patient will die within a year of discharge in a binary fashion, models can be trained to also predict when each patient will die within the year, if at all. This would require the use of more complex model types that can consider time, but would result in predictions that are more interpretable, as the severity of a patient's situations could be assessed using their predicted date of death.

References

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953

Cross, J. L., Choma, M. A., & Onofrey, J. A. (2024). Bias in medical AI: Implications for clinical decision-making. *PLOS digital health*, *3*(11), e0000651. https://doi.org/10.1371/journal.pdig.0000651

Gao, Z., Liu, X., Kang, Y., Hu, P., Zhang, X., Yan, W., Yan, M., Yu, P., Zhang, Q., Xiao, W., & Zhang, Z. (2024). Improving the Prognostic Evaluation Precision of Hospital Outcomes for Heart Failure Using Admission Notes and Clinical Tabular Data: Multimodal Deep Learning Model. *Journal of medical Internet research*, *26*, e54363. https://doi.org/10.2196/54363

Garriga, R., Buda, T. S., Guerreiro, J., Omaña Iglesias, J., Estella Aguerri, I., & Matić, A. (2023). Combining clinical notes with structured electronic health records enhances the prediction of mental health crises. *Cell reports. Medicine*, *4*(11), 101260. https://doi.org/10.1016/j.xcrm.2023.101260

Huang, K., Gray, T. F., Romero-Brufau, S., Tulsky, J. A., & Lindvall, C. (2021). Using nursing notes to improve clinical outcome prediction in intensive care patients: A retrospective cohort study. *Journal of the American Medical Informatics Association : JAMIA*, *28*(8), 1660–1666. https://doi.org/10.1093/jamia/ocab051

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2023). *An introduction to statistical learning: With applications in R* (2nd ed.). Springer. https://doi.org/10.1007/978-1-0716-1418-1

Johnson, A., Bulgarelli, L., Pollard, T., Gow, B., Moody, B., Horng, S., Celi, L. A., & Mark, R. (2024). *MIMIC-IV*. PhysioNet. Retrieved March 2, 2025, from https://physionet.org/content/mimiciv/3.1/icu/#files-panel

Johnson, A., Pollard, T., Horng, S., Celi, L. A., & Mark, R. (2023, January 6). *MIMIC-IV-Note: Deidentified free-text clinical notes*. PhysioNet. Retrieved March 22, 2025, from https://physionet.org/content/mimic-iv-note/2.2/

Olusegun, J. (2023). Historical overview of data analytics in the medical field (*Unpublished manuscript*). Retrieved from https://www.researchgate.net/publication/386170250_Historical_Overview_of_Data_Analytics_in_the_Medical_Field

Petersson, L., Vincent, K., Svedberg, P., Nygren, J. M., & Larsson, I. (2023). Ethical considerations in implementing AI for mortality prediction in the emergency department: Linking theory and practice. *DIGITAL HEALTH, 9*. https://doi.org/10.1177/20552076231206588

Rosenbloom, S. T., Denny, J. C., Xu, H., Lorenzi, N., Stead, W. W., & Johnson, K. B. (2011). Data from clinical notes: a perspective on the tension between structure and flexible documentation. *Journal of the American Medical Informatics Association : JAMIA*, *18*(2), 181–186. https://doi.org/10.1136/jamia.2010.007237

Rosenbloom, S. T., Stead, W. W., Denny, J. C., Giuse, D., Lorenzi, N. M., Brown, S. H., & Johnson, K. B. (2010). Generating Clinical Notes for Electronic Health Record Systems. *Applied clinical informatics*, *1*(3), 232–243. https://doi.org/10.4338/ACI-2010-03-RA-0019

Song, J., Hobensack, M., Bowles, K. H., McDonald, M. V., Cato, K., Rossetti, S. C., Chae, S., Kennedy, E., Barrón, Y., Sridharan, S., & Topaz, M. (2022). Clinical notes: An untapped opportunity for improving risk prediction for hospitalization and emergency department visit during home health care, *Journal of Biomedical Informatics*, 128, 104039, https://doi.org/10.1016/j.jbi.2022.104039.

Toma, M., & Wei, O. C. (2023). Predictive Modeling in Medicine. *Encyclopedia*, *3*(2), 590-601. https://doi.org/10.3390/encyclopedia3020042

Uwinama, I., Rusanganwa , J., & Ingabire, E. (2023). The Moral Implications of Big Data and Machine Learning in Healthcare: A Review. *American Journal of Technology*, *2*(1), 45–53. Retrieved from https://gprjournals.org/journals/index.php/AJT/article/view/145

Zhang Z. (2016). A gentle introduction to artificial neural networks. *Annals of translational medicine*, *4*(19), 370. https://doi.org/10.21037/atm.2016.06.20

Wilson, D. L. (1972). Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC-2*(3), 408–421. https://doi.org/10.1109/TSMC.1972.4309137

Appendix

Table A1. A subset of the final output obtained using bag of words.

acute	bid	blood	care	chest	clear	continued	course	ct	daily	...	refill	right	service	sig	sp	tablet	time	unit	wbc	week
1	2	1	2	0	3	0	2	0	12	...	1	0	1	0	0	2	2	1	2	7
3	7	13	1	1	5	1	1	0	14	...	0	0	1	0	1	0	1	1	3	0
6	8	14	6	1	3	4	5	4	6	...	0	0	1	0	0	0	0	1	2	0
1	8	11	2	0	1	3	1	0	7	...	1	2	2	0	0	3	1	1	4	0
5	0	15	1	3	3	1	0	1	7	...	2	0	2	0	3	2	1	1	2	1

Table A2. A subset of the final output obtained using binary bag of words.

admitted	alert	allergy	attending	birth	blood	brief	chief	clear	complaint	...	procedure	rbc	rdw	result	service	social	surgical	time	unit	wbc
0	1	1	1	1	1	1	1	1	1	...	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	...	1	0	0	1	1	1	1	1	1	1
1	1	1	1	1	1	0	1	1	1	...	1	1	1	1	1	1	1	0	1	1
1	1	1	1	1	1	0	1	1	1	...	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	0	1	1	1	...	1	1	1	1	1	1	1	1	1	1

Table A3. A subset of the final output obtained using TF-IDF.

acute	bid	capsule	care	change	chest	chronic	continued	creat	ct	...	sig	sp	stable	started	tablet	take
0.061094	0.107466	0.0	0.112130	0.000000	0.000000	0.000000	0.000000	0.051966	0.000000	...	0.0	0.000000	0.059225	0.000000	0.112058	0.058913
0.172776	0.354570	0.0	0.052851	0.000000	0.055630	0.000000	0.058672	0.000000	0.000000	...	0.0	0.056457	0.055830	0.059815	0.000000	0.055535
0.347589	0.407612	0.0	0.318975	0.056082	0.055958	0.134315	0.236071	0.098551	0.272456	...	0.0	0.000000	0.056159	0.060167	0.000000	0.000000
0.060176	0.423406	0.0	0.110445	0.058255	0.000000	0.069760	0.183913	0.102369	0.000000	...	0.0	0.000000	0.058335	0.062499	0.165561	0.058027
0.263851	0.000000	0.0	0.048426	0.153257	0.152917	0.061174	0.053760	0.089770	0.062046	...	0.0	0.155191	0.000000	0.109614	0.096790	0.050886

Table A4. Results of baseline models with resampling (using hybrid random oversampling and random undersampling), trained using only structured data

	Logistic Regression	Decision Tree	Random Forest	Gradient Boost	Neural Network
Accuracy	0.691	0.673	0.686	0.645	0.701
Precision	0.291	0.283	0.292	0.262	0.299
Recall	0.715	0.741	**0.744**	0.733	0.714
F1	0.414	0.409	0.419	0.386	0.422
AUROC	0.701	0.701	0.71	0.681	0.706

Table A5. The results of the models trained using both the structured data and discharge summaries, processed using four different NLP techniques

		Logistic Regression	Decision Tree	Random Forest	Gradient Boost	Neural Network
Bag of Words	Accuracy	0.728	0.673	0.707	0.714	0.746
	Precision	0.327	0.287	0.312	0.32	0.333
	Recall	0.74	0.768	0.765	**0.778**	0.666
	F1	0.453	0.418	0.443	0.453	0.444
	AUC	0.733	0.712	0.731	0.74	0.713
Binary Bag of Words	Accuracy	0.693	0.673	0.687	0.684	0.702
	Precision	0.293	0.282	0.291	0.292	0.293
	Recall	0.718	0.74	0.735	**0.754**	0.677
	F1	0.416	0.408	0.417	0.421	0.409
	AUC	0.703	0.7	0.707	0.713	0.692
TF-IDF	Accuracy	0.727	0.681	0.717	0.719	0.749
	Precision	0.327	0.291	0.32	0.325	0.338
	Recall	0.751	0.764	0.765	**0.779**	0.673
	F1	0.456	0.422	0.452	0.458	0.45
	AUC	0.737	0.715	0.737	0.744	0.718
Sentiment Analysis	Accuracy	0.691	0.678	0.689	0.688	0.696
	Precision	0.291	0.286	0.294	0.296	0.297
	Recall	0.715	0.741	0.745	**0.759**	0.724
	F1	0.414	0.412	0.422	0.426	0.421
	AUC	0.701	0.704	0.712	0.717	0.708