Predicting Postoperative Outcomes in Heart Transplantation Using Operative Reports

Adrian Steene
ad3122st-s@student.lu.se

Christoffer Sylve
ch0642sy-s@student.lu.se

Abstract

Heart transplantation is a high-risk procedure often associated with significant post-operative complications. While mortality rates vary across regions and patient populations, our data indicates a first-year mortality rate of approximately 7%. This raises the question of whether post-surgical complications and mortality can be predicted using information from previous surgeries.

In this paper, we present techniques and models that combine post-operative reports in text, with structured numerical and categorical data to predict patient mortality at 30 days, 1 year, and 3 years after surgery. Using five-fold cross-validation, our models achieved macro F1 scores of 0.75, 0.64, and 0.61, respectively. We also demonstrate that the same approach can be used to predict post-operative complications and prolonged hospital stays (over 25 days), with macro F1 scores of 0.65 and 0.62.

Despite a relatively small sample size, our findings suggest that clinical narratives—when properly processed—play a significant role in improving the predictive accuracy of short-term post-transplant outcomes.

1 Introduction

Heart transplantation is a life-saving procedure involving the surgical replacement of a failing heart with a donor's healthy heart. As one of the most complex interventions in modern medicine, it carries significant risk to the recipient. Our observed data indicates a first-year post-operative mortality rate of approximately 7%. However, this risk decreases to around 4% annually after the first year, highlighting the initial period as the most critical for patient survival.

Given the shortage of donor hearts relative to the number of recipients on waiting lists, it is crucial to optimize patient selection by assessing individual risk to maximize survival outcomes and post-operative quality of life. In addition to structured numerical and categorical data, a vast amount of information is generated through pre- and post-operative assessments documented by medical professionals. To our knowledge, most predictive analyses rely primarily on structured data, potentially overlooking valuable semantic insights embedded in clinical narratives.

In this report, we demonstrate how post-operative clinical reports can be processed and integrated with structured data to predict patient survival following heart transplantation. Our models achieved macro F1 scores of 0.75, 0.64, and 0.61 for predicting survival at 30 days, 1 year, and 3 years, respectively. Additionally, we show that this approach can be used to predict post-operative complications and prolonged hospital stays (longer than 25 days), with macro F1 scores of 0.65 and 0.62. All data used in this study were anonymized and approved for research purposes.

2 Previous work

This study builds upon the work of (Klang et al., 2022), which introduced the use of operative reports in text format for predicting post-transplant survival outcomes. Their research demonstrated the potential of natural language processing techniques to extract clinically relevant information from unstructured text for use in predictive modeling.

While Klang et al. primarily focused on text-based features and employed neural network models, our work extends this in several important ways. First, we leverage a larger and more comprehensive dataset that includes both unstructured clinical text and structured numerical values, allowing us to explore the effectiveness of combining multiple data modalities.

Second, we adopt logistic regression as our primary modeling technique, which is less computa-

tionally intensive and more interpretable than neural networks. This enables us to focus on enhancing model performance through systematic hyperparameter tuning using grid search.

Finally, we expanded the scope of predictive features beyond long-term survival. Insted of focusing solely on 1- and 5-year mortality, we opted for 30 days, 1 year and 3 years which puts more focus of the short term outcomes. We also extended the analysis to include post-operative complications and prolonged hospitalization (defined as stays longer than 25 days).

3 Materials and methods

3.1 Corpus

The corpus used in this study contains a collection of data from 472 patients who underwent heart transplant surgery. For each patient, three Swedish text features were available. The three features were Opber, Diagnoser and Opkoder. Below is a short extract from the three text features:

Opber. Postoperativt har han successivt återhämtat sig från hjärtsvikt och har kunnat mobiliseras tillräckligt...

Diagnoser. Status postop HeartMate LVAD för dilaterad kardiomyopati I42.0 _x000d_Post hjärttransplantation hjärtsvikt I97.1 _x000d_

Opkoder. Ortotop hjärttransplantation med bikaval anastomos FQA10 _x000d_Total kardiopulmonell bypass i moderat hypotermi FXA00...

The Opber corpus is the largest corpus in the dataset and it contained about 143,500 words, with the most common being och (and), med (with), i (in), på (on) och hjärtat (heart). The corpus contained approximately 10,400 unique lowercased words.

The Diagnoser corpus contained approximately 4,800 words, with 550 unique lowercased tokens. The Opkoder corpus contained around 7,700 words, with 330 unique lowercased tokens.

The dataset also contained 2 numeric features: sex, age. Sex is a binary feature encoded as 0 for female and 1 for male. Age is a continuous variable representing patient age. For each patient, the number of words in the Opber feature was calculated and added as a new numeric feature named num_words.

The dataset also contained 8 binary features for different post-operative complications. Seven of these were selected and a new feature introduced called had_compl. This feature is equal to 1 if the patient experienced at least one of the selected complications, and 0 otherwise.

Finally, the dataset also contained a label LOS (Length of stay), representing the number of days each patient spent at the hospital post-transplant. A new feature was introduced called stayed_more_than_25_days, the threshold was based on the distribution of LOS values.

Table 1: Summary Statistics for Target features

Label	N	Positive	Negative
30-day mortality	470	453	17
1-year mortality	453	420	33
3-year mortality	407	355	52
`had_compl`	471	182	289
`stayed_>25_d`	471	141	330

3.2 Algorithms

Logistic regression was used with TF-IDF (Term Frequency–Inverse Document Frequency) vectorization(Pradeep, 2023). This method assigns a value to each token based on how frequently it appears in the current document relative to its frequency across all documents. A high TF-IDF value indicates that a word is particularly informative, as it occurs infrequently in other documents. For the numeric features, we standardized them, so they do not gain a disproportionately large weight(scikit-learn developers, 2025b). Other models like Random Forest were evaluated, but given the high density and sparse data produced from TF-IDF vectorization, logistic regression was considered the best choice.

4 Experimental Setup

We preprocessed Opber by removing punctuation and digits and converting all text to lowercase. The preprocessing was not applied to Diagnoser and Opkoder because the digits in these features are a code that could prove useful for our model. For all text features, we used TF-IDF vectorization from sklearn (scikit-learn developers, 2025c). We used a StandardScaler from sklearn for all the numeric features. We ran all our tests on a 5 fold cross-validation. Given the dataset's highly imbalanced class distribution, we applied random oversampling to increase the minority class. To optimize the models hyperparameter we performed a grid search

using the sklearn GridSearchCV(scikit-learn developers, 2025a). Grid search was performed independently for each run to find the combinations that yielded the highest cross-validation F1 macro score. Below is the serach space:

- **TF-IDF parameters:** maximum number of features `[50, 200, 500]` and n-gram ranges `(1,1)` and `(1,2)`
- **Oversampling ratio:** sampling strategies between `0.1` and `0.5`.
- **Logistic regression:** regularization strength `C` in `[0.01, 0.1, 1.0, 10.0, 30]`, class weighting options `[None, balanced]`, and iteration limits `100–900`.

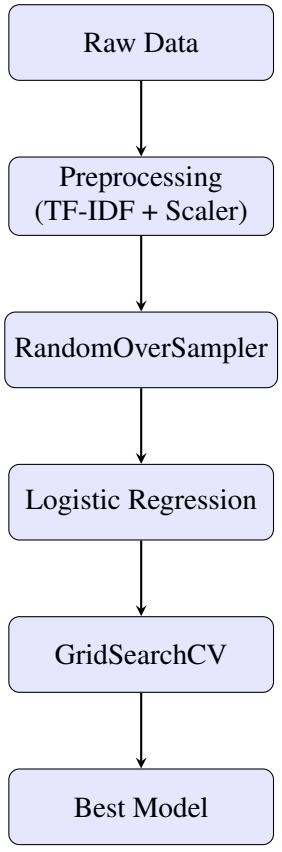

Figure 1: Pipeline

5 Results

For scoring we used F1 macro, which takes the average F1 score for both classes and weights them equally(Wikipedia contributors, 2025). This is a good choice due to our dataset being imbalanced, using just the standard F1 score would largely ignore the minority class. ROC AUC (Receiver Operating Characteristic - Area Under Curve) measures a model's ability to rank positive cases higher than negative(GeeksforGeeks contributors, 2021). It uses the output of the model probability, which means that even if the final classification is wrong, strong ranking performance can still be reflected.

Mortality 30 days had the best results of all models, reaching an F1 macro score of 0.749 with Opber and Opkoder. This suggests that for early mortality outcomes, Opkoder has valuable information. The best ROC-AUC was 0.824 for the model with Opber and age, but with a relatively low macro-F1 score of 0.673. This indicates that the model is good at identifying high-risk patients, the low F1 macro suggests that the threshold function is suboptimal.

Mortality 1 year had a generally lower score than mortality 30 days. This is likely due to the operation itself being less important after 1 year. The best F1 macro score was 0.638 using Opber, Opkoder, num_words, age, sex. This indicates that the model has a harder time distinguishing the different classes and adding more general information about the patient helps the model. The best ROC-AUC score was achieved with Opber, Opkoder, num_words, age, diagnoser, sex. Also here a combination of text and numeric features proves to be the best. Similarly, for mortality at 30 days, the ROC-AUC score is higher than the F1 macro-score, indicating a suboptimal threshold function.

For Mortality 3 years the model shows a similar pattern, relatively low F1 macro score but a moderate ROC-AUC score. The combination Opber, Opkoder and Diagnoser produced the best model with an F1 macro of 0.618. This indicates that Diagnoser adds information for long-time survival. The highest ROC-AUC score was with the combination Opber, Opkoder, Diagnoser and age. Age seems to improve the ROC-AUC score, which is intuitive that after 3 years of the operation the operation itself is less important, it is rather the patient's general health that is important.

To predict whether the LOS would be for 25 days, the model performed moderately well with a macro F1 score of 0.647 for Opber alone. Adding features did not improve the model performance, which indicates that the report itself is a moderately good predictor for LOS.

Predicting if a patient would have a postoperative complication was less successful than LOS. An F1 macro score of 0.617, with the features Opber and Opkoder was the best result.

Table 2: Performance comparison between the Opber-only baseline and the best-performing feature combination for each prediction task.

Prediction Task	Feature Set	F1 (Macro)	ROC AUC
30-day Mortality	Opber only	0.707	0.752
	Opber + Opkoder (Best)	**0.749**	**0.772**
1-year Mortality	Opber only	0.572	0.744
	Opber + Opkoder + Num_words + Age + Sex (Best)	**0.638**	**0.789**
3-year Mortality	Opber only	0.587	0.670
	Opber + Opkoder + Age (Best)	**0.609**	**0.696**
LOS < 25 days	Opber only	0.589	0.656
	Opber + Num_words (Best)	**0.647**	**0.708**
Had Complication	Opber only	0.586	0.603
	Opber + Opkoder (Best)	**0.617**	**0.616**

6 Conclusion

The results for the features vary, but are generally promising. The best results were achieved for Mortality 30 days, this indicates that the operation report is especially important for short-term survival. For mortality 1 and 3 years the results are lower, this is to be expected as the immediate impact of surgery diminishes over time. Adding features to the model improved results, which indicates that more information is needed to predict the outcome. For LOS and had complication the results are similar to mortality 1 and 3 years, the difference being that adding features did not improve the model. Our paper shows that the text it self is a strong indicater on patient outcome. To improve results a bigger dataset would be welcomed, also getting access to a Swedish bert trained on medicine would be very interesting to application.

References

GeeksforGeeks contributors. 2021. Auc - roc curve in machine learning. https://www.geeksforgeeks.org/auc-roc-curve/. Accessed: 2025-06-09.

Marcus Klang, Daniel Diaz, Dennis Medved, Pierre Nugues, and Johan Nilsson. 2022. Using operative reports to predict heart transplantation survival. In *2022 44th Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC)*, pages 2258–2261.

Er. Pradeep. 2023. Understanding tf-idf in nlp: A comprehensive guide. https://medium.com/@er.iit.pradeep09/understanding-tf-idf-in-nlp-a-comprehensive-guide-26707db0cec5. Accessed: 2025-06-09.

scikit-learn developers. 2025a. GridSearchCV — scikit-learn documentation. Accessed: 2025-06-09.

scikit-learn developers. 2025b. LogisticRegression — scikit-learn documentation. Accessed: 2025-06-09.

scikit-learn developers. 2025c. TfidfVectorizer — scikit-learn documentation. Accessed: 2025-06-09.

Wikipedia contributors. 2025. F-score – macro f1. https://en.wikipedia.org/wiki/F-score#Macro_F1. Accessed: 2025-06-09.

7 Appendix

Table 3: Model performance by feature combination with label mortality 30 days.

Features Used	**F1 Macro**	**ROC AUC**
Opber	0.719	0.802
Opber + Diagnoser	0.678	0.806
Opber + Opkoder	**0.749**	0.772
Opber + Sex	0.719	0.789
Opber + age	0.673	**0.824**
Opber + num_words	0.702	0.767
Opber + opkoder + sex	0.735	0.761
Opber + opkoder + age	0.712	0.849
Opber + opkoder + num_words	0.719	0.774
Opber + opkoder + diagnoser	0.678	0.798

Table 4: Model performance by feature combination with label mortality 1 year.

Features Used	**F1 Macro**	**ROC AUC**
Opber	0.613	0.592
Opber + diagnoser	0.621	0.542
Opber + opkoder	**6**0.622	0.609
Opber + sex	0.595	0.598
Opber + age	0.610	**0.637**
Opber + num_words	0.621	0.613
Opber + opkoder + sex	0.618	0.555
Opber + opkoder + age	0.616	**0.748**
Opber + opkoder + num_words	**0.631**	0.601
Opber + opkoder + diagnoser	0.616	0.614
Opber + opkoder + num_words + sex	0.614	0.591
Opber + opkoder + num_words + age	**0.633**	**0.770**
Opber + opkoder + num_words + diagnoser	0.6132	0.6936
Opber + opkoder + num_words + age + sex	**0.638**	0.733
Opber + opkoder + num_words + age + diagnoser	0.602	0.712
Opber + opkoder + num_words + age + diagnoser + sex	0.600	**0.789**

Table 5: Model performance by feature combination with label mortality 3 year.

Feature Combination	**F1 Macro**	**ROC AUC**
Opber	0.566	0.581
Opber + Diagnoser	0.583	0.606
Opber + Opkoder	**0.601**	0.562
Opber + SEX	0.538	0.603
Opber + age	0.550	**0.666**
Opber + num_words	0.525	0.598
Opber + opkoder + sex	0.578	0.610
Opber + opkoder + age	0.609	**0.696**
Opber + opkoder + num_words	0.581	0.585
Opber + opkoder + diagnoser	**0.618**	0.555
Opber + opkoder + diagnoser + sex	0.59	0.57
Opber + opkoder + diagnoser + age	0.59	0.72
Opber + opkoder + diagnoser + num_words	0.59	0.59

Table 6: Model performance by feature combination with label LOS less than 25 days.

Features Used	**F1 Macro**	**ROC AUC**
Opber	0.647	0.711
Opber + Diagnoser	0.619	0.677
Opber + Opkoder	0.636	0.676
Opber + op_year	0.628	0.693
Opber + SEX	0.628	0.681
Opber + age	0.594	**0.727**
Opber + num_words	**0.647**	0.708
Opber + num_words + SEX	0.629	0.698
Opber + num_words + age	0.595	0.726
Opber + num_words + Diagnoser	0.618	0.686
Opber + num_words + Opkoder	0.638	0.672

Table 7: Model performance by feature combination with label had complication

Features Used	**F1 Macro**	**ROC AUC**
Opber	0.580	0.606
Opber + sex	0.579	0.592
Opber + age	0.582	0.639
Opber + num_words	0.574	0.608
Opber + diagnoser	0.606	0.606
Opber + opkoder	**0.617**	**0.616**
Opber + opkoder + sex	0.613	0.611
Opber + opkoder + age	0.613	0.611
Opber + opkoder + num_words	0.613	0.610
Opber + opkoder + diagnoser	0.606	0.611