MSI608 - Project #1, July 2025

Digital Object Identifier

DRG code prediction and data analysis using ML model

GENESIS PIÑA, MARCELO ROZAS, CARLOS FUENTES, RICARDO MEDINA
Faculty of Engineering, Master's Degree in Computer Engineering, Universidad Andrés Bello

This work was developed as part of the course MSI608: Advanced Topics in Data Science.

ABSTRACT Identification of the diagnosis-related group (DRG) is essential for hospital resource planning. This study proposes the analysis of data from El Pino Hospital, and creates and evaluates a ML model to predict the DRG code of a patient upon admission, using diagnosis and procedure data. Our goal is to optimize early resource allocation and improve hospital logistics.

INDEX TERMS DRG, neural networks, hospital logistics, healthcare prediction, machine learning, classification

I. INTRODUCTION

The Diagnostic Group System (DRG) is widely used in hospital systems to classify patients based on clinical characteristics and expected use of resources. Each DRG code comprises a main diagnostic category and a severity index ranging from 1 (mild) to 3 (severe). This classification supports standardized clinical management, hospital planning, and resource allocation.

Our study aims to predict the DRG code of a patient'using a data set from El Pino Hospital. The data set includes diagnostic codes, procedures, age, and sex. Predicting early DRG can improve clinical planning, risk assessment, and economic forecasting in the hospital setting.

Predicting DRG automatically and early in the patient's treatment has several benefits:

- Improves the management of hospital beds, resources, and surgical schedules.
- Facilitates the assessment of patient risk and severity.
- Allows for the identification of severe cases in the early stages.
- Supports financial and insurance coverage decision making.

In this study, we specifically address the use of predictive models for DRG, utilizing neural networks, as a strategy to improve hospital logistics management, beyond their traditional application in financial and post-discharge contexts. The ability to accurately anticipate a patient's DRG upon admission would allow the hospital to make informed decisions in advance regarding the following:

- The assignment of hospital beds and rooms according to the expected severity.
- The scheduling of shifts of clinical personnel, based on projected care workload.
- Management of stock and distribution of medical supplies and medications, according to the type of care required.
- Prioritization of the use of operating rooms, critical care beds, or specialized units.

II. REFERENCES TO SIMILAR PROBLEMS

The problem of predicting a patient's DRG has been addressed in various studies, both from the perspective of medical informatics and hospital management, due to its importance in improving healthcare system efficiency. In particular, practical evidence shows the use of machine learning and deep learning techniques to estimate both the diagnostic category and the severity level of DRGs, using structured data from electronic health records (EHRs), prior diagnoses, procedures, and other relevant clinical variables.

A. *WHAT TECHNIQUES HAVE BEEN USED TO ADDRESS IT?*

The most common approaches to tackle this problem include supervised classification models such as logistic regression, decision trees, and Random Forest (Bertsimas et al., 2008), as well as ensemble learning techniques like XGBoost (Baek et al., 2021). More recently, deep neural networks (DNNs), recurrent networks (RNNs and LSTMs) for medical event time sequences (Jiang et al., 2023), and interpretable architectures such as RETAIN (Reverse Time Attention Model) have been incorporated, combining strong performance with clinical explainability (Choi et al., 2016).

B. WHAT RESULTS HAVE BEEN OBTAINED?

Recent studies report promising results. For example, Baek et al. (2021) developed XGBoost-based models to predict DRG with more precision 80%, highlighting the predictive value of diagnoses and surgical procedures. Jiang et al. (2023) applied deep learning to EHRs and achieved highly accurate predictions, even when distinguishing between levels of clinical severity. Models like RETAIN (Choi et al., 2016) have proven useful by offering a clear interpretation of the most relevant clinical factors. Finally, Bertsimas et al. (2008) demonstrated that data mining techniques could outperform traditional DRG classification models in terms of accuracy and administrative applicability.

These studies support the relevance of the problem and show that there are proven techniques that could be adapted to the context of Hospital El Pino, considering the characteristics of our data set.

REFERENCES

[1] H. Baek et al., "Development and validation of machine learning models for predicting Diagnosis Related Groups (DRGs)," *J. Biomedical Informatics*, vol. 115, 2021.

[2] T. Jiang et al., "Deep learning for severity prediction in DRG classification using EHR data," *Artificial Intelligence in Medicine*, vol. 138, 2023.

[3] E. Choi et al., "RETAIN: An interpretable predictive model for healthcare using reverse time attention mechanism," in *NeurIPS*, 2016.

[4] D. Bertsimas et al., "Algorithmic prediction of health-care costs," *Operations Research*, vol. 56, no. 6, 2008.

III. OBJECTIVES

This study aims to develop a neural network-based predictive model to estimate the DRG code for a patient so the hospital can have this information as early as possible, supporting key logistical decisions that can improve system efficiency, reduce waiting times, prevent overloads, and improve the quality of care provided to hospital patients.

The main objective of this study is to develop a predictive model based on neural networks that can accurately anticipate the DRG code of a patient as early as possible, using the set of recorded diagnoses and procedures as input variables.

The early prediction will allow Hospital El Pino to support key logistical decision-making, such as the allocation of human and material resources, the planning of medical staff shifts, the efficient distribution of beds and operating rooms, and the proactive management of clinical supplies, according to the expected severity of the case.

Specific objectives include:

- Implement a neural network architecture suitable for multi-class classification.
- Evaluate the model using accuracy, precision, recall, and F1-score.
- Analyze the model's ability to distinguish among DRG severity levels.
- Identify patterns in diagnosis and procedures that correlate with high-severity cases.

This study aims not only to validate the technical feasibility of the model within the context of the provided dataset, but also to demonstrate its practical usefulness as a tool for supporting predictive hospital management—especially in institutions with a high patient load, such as Hospital El Pino in this case of study.

IV. METHODOLOGY

A. DATASET DESCRIPTION

The dataset contains 14,561 records (patients) and 68 columns (features), the are the following:

- 35 diagnosis codes
  - Each column is **categorical**, with a code (e.x. A41.8) and a description.
  - The diagnosis codes follow IDC-10 standard
- 30 procedure codes
  - Each column is **categorical**, with a numerical code (e.x. 93.01) and a description.
  - The procedure codes follow IDC-9 standard
- 2 demo variables
  - Age, **numerical** feature
  - Sex, **categorical** feature
- Target variable: **DRG code**
  - Code composed of a diagnostic prefix and a severity suffix (1 = mild, 2 = moderate, 3 = severe). Example: 184103 = MH SEPTICEMIA WMCC or Severe Septicemia.
  - There are no cases with severity 0

In total, there are 67 independent variables (35 diagnoses + 30 procedures + Age + Sex) and 1 dependent variable (DRG). At this point we can define the problem a multiclass classification task (there are several DRG codes), with an imbalanced distribution as we will see next.

B. EXPLORATORY DATA ANALYSIS

The step we are taking are as follow:

- Loading and copying of the CSV (semi-colon delimited) file generated from file "dataset_elpino.csv".
- Renaming of columns to simplified names andlower case. For example:
  - "Diag 01 Principal (cod+des)" → diag_01
  - "Diag 02 Secundario (cod+des)" → diag_02
  - ...
  - Proced 01 Principal (cod+des) → proc_01
  - Proced 02 Secundario (cod+des) → proc_02
  - ...
- Verification of shape (df.shape) and data types (df.dtypes)
  - Rows → 14,561
  - Columns → 68
- Detection and handling of missing values (if any).
  - There are no NULL values, but we can see a dash '-' when there is no diagnosis or procedure.
  - For columns: diag_01, proc_01, edad, sexo and grd there are no missing values, so we can say all

patient have at least 1 diagnosis and 1 procedure along with demo information and grd.
  - Diagnosis and procedure columns are increasing missing values as their index increase, this makes sense since it holds up to 35 diagnosis and 30 procedures but a patient can has less than that, even 1.
  - Given this, we are not going to impute missing values.
- Check duplicate records.
  - There are 144 duplicate records, most of them are having age of 0, meaning it is a new born children with only 1 diagnosis and 1 procedure, so it makes sense there are several of those.
  - There are 12 duplicates with age > 0, most of them related to labor and delivery care, so it makes sense also to have such duplicates.
  - Given this, we will not remove or treat duplicates.
- Frequency by sex (Hombre/Mujer). Figure 1

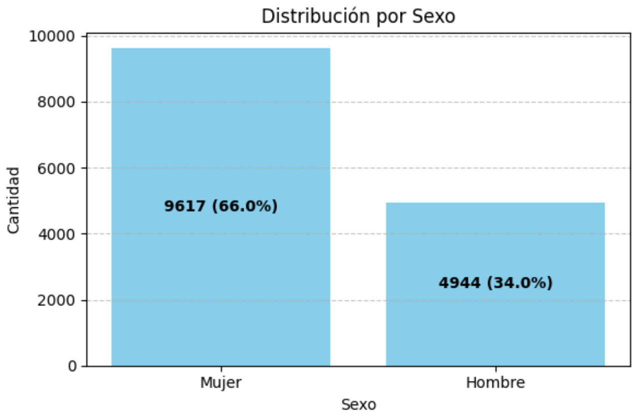


FIGURE 1. Sex Distribution

- Frequency by age. Figure 2

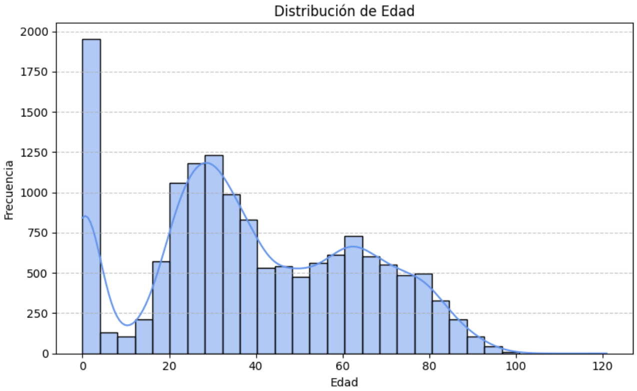
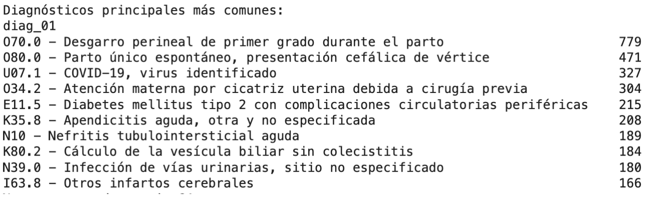


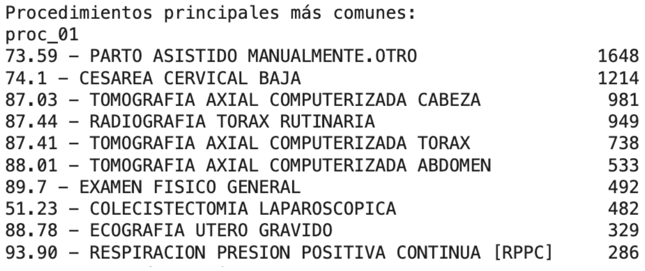
FIGURE 2. Age Distribution

- Most common diagnosis with frequency. Figure 3
- Most common procedures with frequency. Figure 4
- Most common DRG with frequency. Figure 5



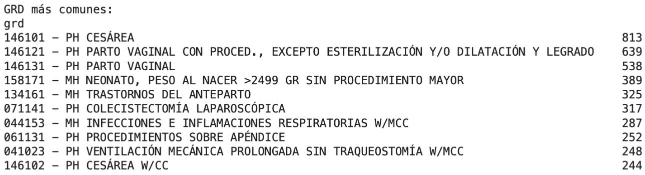
```
Diagnósticos principales más comunes:
diag_01
O70.0 – Desgarro perineal de primer grado durante el parto              779
O80.0 – Parto único espontáneo, presentación cefálica de vértice        471
U07.1 – COVID–19, virus identificado                                    327
O34.2 – Atención materna por cicatriz uterina debida a cirugía previa   304
E11.5 – Diabetes mellitus tipo 2 con complicaciones circulatorias periféricas   215
K35.8 – Apendicitis aguda, otra y no especificada                       208
N10 – Nefritis tubulointersticial aguda                                 189
K80.2 – Cálculo de la vesícula biliar sin colecistitis                  184
N39.0 – Infección de vías urinarias, sitio no especificado              180
I63.8 – Otros infartos cerebrales                                       166
```

FIGURE 3. Top freq diagnosis



```
Procedimientos principales más comunes:
proc_01
73.59 – PARTO ASISTIDO MANUALMENTE.OTRO                 1648
74.1 – CESAREA CERVICAL BAJA                            1214
87.03 – TOMOGRAFIA AXIAL COMPUTERIZADA CABEZA            981
87.44 – RADIOGRAFIA TORAX RUTINARIA                      949
87.41 – TOMOGRAFIA AXIAL COMPUTERIZADA TORAX             738
88.01 – TOMOGRAFIA AXIAL COMPUTERIZADA ABDOMEN           533
89.7 – EXAMEN FISICO GENERAL                             492
51.23 – COLECISTECTOMIA LAPAROSCOPICA                    482
88.78 – ECOGRAFIA UTERO GRAVIDO                          329
93.90 – RESPIRACION PRESION POSITIVA CONTINUA [RPPC]     286
```

FIGURE 4. Top freq Procedure



```
GRD más comunes:
grd
146101 – PH CESÁREA                                                                          813
146121 – PH PARTO VAGINAL CON PROCED., EXCEPTO ESTERILIZACIÓN Y/O DILATACIÓN Y LEGRADO      639
146131 – PH PARTO VAGINAL                                                                    538
158171 – MH NEONATO, PESO AL NACER >2499 GR SIN PROCEDIMIENTO MAYOR                          389
134161 – MH TRASTORNOS DEL ANTEPARTO                                                         325
071141 – PH COLECISTECTOMÍA LAPAROSCÓPICA                                                    317
044153 – MH INFECCIONES E INFLAMACIONES RESPIRATORIAS W/MCC                                  287
061131 – PH PROCEDIMIENTOS SOBRE APÉNDICE                                                    252
041023 – PH VENTILACIÓN MECÁNICA PROLONGADA SIN TRAQUEOSTOMÍA W/MCC                          248
146102 – PH CESÁREA W/CC                                                                     244
```

FIGURE 5. Top freq DRG

- Given that diagnosis, procedures and drg are composed of code description, we separate both so we can continue working in an easier way only with codes. Columns having codes will end with '_cod'.
- Now we split variable grd_cod into two new variables, like this
  - grd_base (diagnostic prefix)
  - grd_level (last digit → severity level 1, 2, or 3)
- Distribution by DRG level. Figure 6

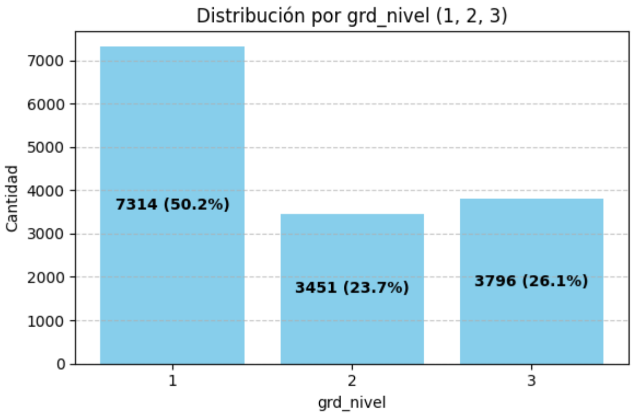


FIGURE 6. DRG level distribution

- Before splitting DRG we had 526 different values, after splitting we have 210 DRG base values, this will make the model to behave better since categories to predict are less (even though still high number).

- For DRG base frequency with cumulative frequency function. Here it is important to note the several os the DRG base codes have a low frequency, even there are many of them with only 1 occurrence (76 of them with only 1 value). Figure 7

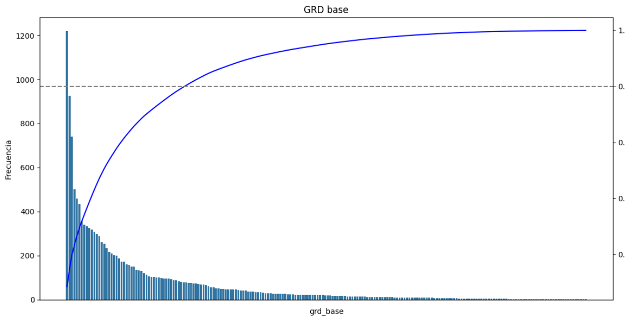


FIGURE 7. DRG level distribution

C. DATA PREPARATION

We have done some data preparation already which has been explained in previous section. Here we will apply steps to prepare data for the model which mainly include data encoding and data separation in train and validation. Here more details on the steps.

- Select the necessary columns, we will use 'cod' columns only (excluding columns having description). They are 68 columns (35 diag, 30 proc, sex, age + dependent variable: grd_base).
- Encoding categorical variables, for this we use: **LabelEncoder**.
- Sex variable transform like this: 'Mujer': 0, 'Hombre': 1.
- Separate columns like: input (X) and output (Y = grd_base).
- Separate data in train (0.8) and validation (0.2).

D. MODEL DEVELOPMENT

In this stage we create the Model based on TensorFlow Keras. We will follow a similar approach like we have seen during the course (MSI-608) where we provide a function to create the model and then we use Keras tuner random search to test different Model hyper-parameters, then the search will deliver the best model which we can use for validation. Here are the steps followed to create the model:

- Since we are using input data with LabelEncoder for categorical variable, we apply 2 transformation: Embedding + Flatten. With the Embedding layer we get a dense vector representation with the following benefits:
  - Compact, learnable representations
  - Efficiency, given we have high amount of features
  - Captures relationships
- It is important to make a note at this point of a Model design decision. We have not used One-Hot encoding because we have a huge amount of different values for the variable to be predicted (grd_base). Give this high amount, the dimensionality of the problem would increase a lot, making the model more complex.
- For the output dimension of the Embedding we are using Choice with the following sizes: 4, 8, 16.
- Model Input Layer: 1 neuron (Input) for each of the variables in the input.
- Model hidden Layer:
  - Embedding (1 for each neuron in the input)
  - Flatten (1 for each Embedding)
  - 2 Dense with **Relu** activation
- Model Output Layer: Dense with the length of values in grd_base, and activation **Softmax**.
- Model compile:
  - optimizer: Adam
  - loss: sparse_categorical_crossentropy
  - metrics: accuracy
- Model tuner:
  - This is keras_tuner.RandomSearch
  - Objective("val_accuracy", direction="max")
  - max_trials: 8,
  - executions_per_trial: 1,
- Model tuner stats:
  - Best val_accuracy So Far: 0.8510127067565918
  - Total elapsed time: 00h 09m 25s
- Best Model stats:
  - Total params: 419,926 (1.60 MB)
  - Trainable params: 419,926 (1.60 MB)
  - Non-trainable params: 0 (0.00 B)
- Best Model Visualization:
  - Figure 8 shows layers Input + Embedding + Flattern. This is one for each variable in the input (67)
  - Figure 9 shows layers Concatenate + Dense + Dense 1 + Dense 2 (210 outputs)

E. EVALUATION METRICS

The evaluation of the model is difficult for this problem because we have a big amount of classes (210). It would be a good approuch to reduce the number of classes available by removing the ones having low frequency, for example <5, but becuase of time constraints we will continue the evaluation of the model as-is with the 210 classes we have available. Here are the numbers we are getting globally from the model evaluation:

- Accuracy: **0.8510**
- Precision (weighted): **0.8738**
- Recall (weighted): **0.8510**
- F1-score (weighted): **0.8533**

The confusion matrix we got from the code is too dense to add it here, again because of the big number of classes, so it will not be added.

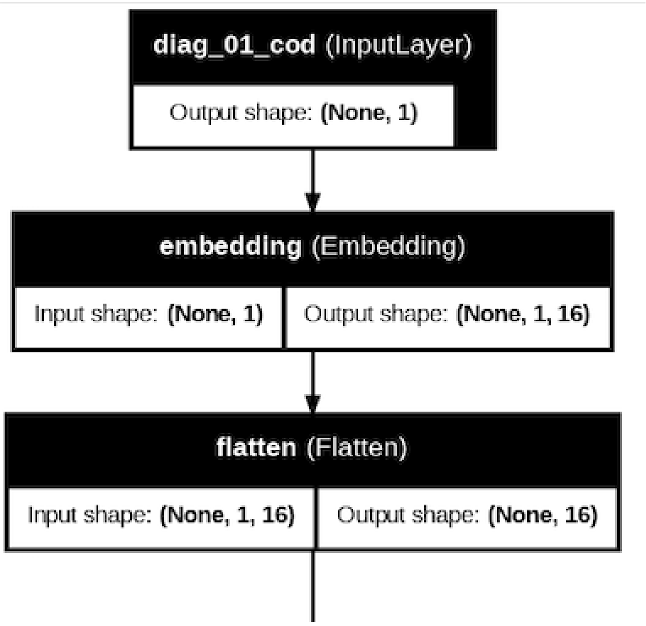


FIGURE 8. Layers: Input + Embedding + Flattern

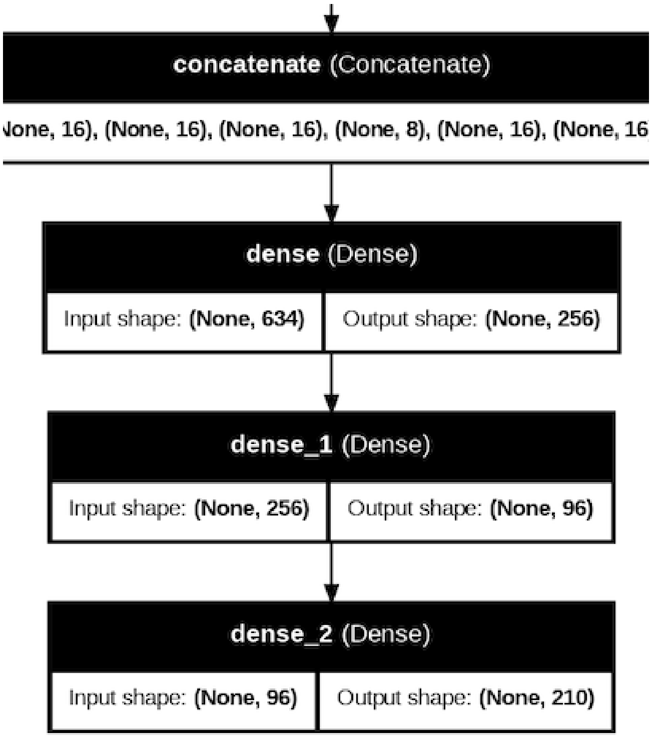

concatenate (Concatenate)
Vone, 16), (None, 16), (None, 16), (None, 8), (None, 16), (None, 16
dense (Dense)
Input shape: (None, 634)
Output shape: (None, 256)
dense_1 (Dense)
Input shape: (None, 256)
Output shape: (None, 96)
dense_2 (Dense)
Input shape: (None, 96)
Output shape: (None, 210)

FIGURE 9. Layers: Concatenate + Dense + Dense 1 + Dense 2 (210 outputs)

Table 1 shows the evaluation metric for the top-5 classes (we are showing index only, but it can be converted back to the base DRG).

Figure 10 shows the ROC curve One-vs-Rest for top-5 classes. As we can see, the accuracy is very high for the top-5 clases.

index	precision	recall	f1-score	support
14610	0.996	1.000	0.998	225
14612	0.961	0.994	0.977	173
14613	0.967	1.000	0.983	148
07114	0.960	0.975	0.967	122
15817	0.744	0.879	0.806	099

TABLE 1. Evaluation metrics for top-5 classes

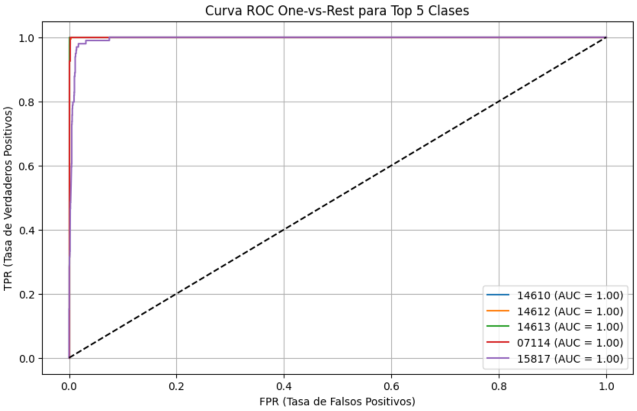


FIGURE 10. ROC curve One-vs-Rest for top-5 classes

V. CONCLUSION

This study presents a predictive model for the severity level of DRG using hospital admission data. Using a neural network approach, the model aims to support early hospital resource planning. The model was trained on structured data from El Pino Hospital, including diagnoses, procedures, sex, and age, and achieved promising results with a validation precision greater than 85

The embedded architecture proved to be effective in handling categorical variables of high cardiac intensity while keeping the size of the model manageable. The separation of the DRG code into diagnostic base and severity level allowed us to reduce class imbalance and improve classification performance.

These findings reinforce the feasibility of implementing data-driven tools in the early stages of patient admission to enhance hospital logistics. Early DRG prediction can improve planning for bed occupancy, staff allocation, and resource usage especially in hight-demand public healthcare settings.

Ultimately, predictive models like the one developed in this study could become part of intelligent hospital infrastructure, helping to prioritize care, reduce bottlenecks, and improve patient outcomes.