SAPIENZA
UNIVERSITÀ DI ROMA

Predicting Clinical ICU-related Outcomes from Electronic Health Records: a Two-Stage Deep Learning Approach combining Transformers and Graph Neural Networks

Dipartimento di Ingegneria informatica, automatica e gestionale
Artificial Intelligence and Robotics

Ettore Branca
ID number 1965733

Advisor
Prof. C.Napoli

Co-Advisor
Dr. co-advisor

Academic Year 2025/2026

Thesis defended on XX January 2026
in front of a Board of Examiners composed by:

Prof. ... (chairman)
Prof. ...
Prof. ...
Prof. ...
Prof. ...
Prof. ...
Prof. ...

Predicting Clinical ICU-related Outcomes from Electronic Health Records: a Two-Stage Deep Learning Approach combining Transformers and Graph Neural Networks
Sapienza University of Rome

This thesis has been typeset by LaTeX and the Sapthesis class.

Author's email: branca.1965733@studenti.uniroma1.it

Abstract

Early detection of clinical decline enables for timely admission to the Intensive Care Unit (ICU) and saves lives. In this work, I present a two-stage Deep Learning pipeline designed to predict several clinical outcomes from the MIMIC-IV electronic health records (EHR) database. This approach first aggregates and pre-processes patient admission data, covering demographic and administrative details, early laboratory flags, ICU vital signs, and ICU timing intervals into a unified tabular representation. Then, a Transformer-based 'Clinical State Estimator' is used to produce an embedding for categorical features and to learn both ICU-transfer risk and time-to-ICU predictions. In the second stage, each hospital admission is represented as a node in a heterogeneous admission-centric hypergraph, with edges that capture three complementary relationships: early-stage similarity (K-Nearest-Neighbors in feature space), predicted ICU transfers (connections to a dummy ICU node, weighted by probabilities produced by the first stage), and temporal links between multiple admissions of the same patient. Finally, a multilayer Graph Neural Network (combining GCN/R-GCN, GATv2 and GraphSAGE with JumpingKnowledge concatenation) processes this graph to jointly predict in-hospital mortality, length of stay in the ICU, and discharge location. Experiments on MIMIC-IV show state of the art results for mortality risk prediction (AUC 0.909, Accuracy 0.963), ICU-duration regression (RMSE 54-55 h, MAE 35-36 h), and discharge-location accuracy (0.742). An ablation study confirmed that both the multi-edge graph design and the multitask supervision yield significant gains. This method relies exclusively on standard hospital data, is end-to-end differentiable, and runs in real time on a single GPU, making it suitable for prospective deployment and a promising direction for graph-aware clinical decision support.

Contents

Chapter 1

Introduction

Unrecognized deterioration in hospital wards often results in emergent transfers to the intensive care unit (ICU), a scenario repeatedly associated with higher mortality and longer stays. Multiple studies report that every hour of delay in admission to the ICU increases the odds of death for critically ill adults. Mixed evidence on whether operational delays alone explain outcomes further underscores the value of earlier detection and proactive triage (Cardoso et al. 2011; Churpek et al. 2016; Kiekkas et al. 2022).

Hospitals commonly deploy early warning scores (EWS) that compare the values of a small set of vital signs and lab tests against fixed thresholds. Although simple, inexpensive and broadly adopted, the clinical benefit of these metrics remains debated. Several articles show that EWS are validated with widely differing datasets, endpoints, time windows, and evaluation metrics, and are often digitized without being fully integrated into clinical decision workflows (Bedoya et al. 2019; Fang, Lim, and Balakrishnan 2020; Nagarajah, Krzyzanowska, and Murphy 2022; Wong et al. 2024). Additionally, EWS can miss complex temporal patterns that are instead embedded in the broader electronic health record (EHR) data.

In recent years, there has been a move towards the adoption of digital health record systems in hospitals: in 2015, in the United States, nearly 96% of hospitals had a digital EHR system. Retrospectively collected medical data has increasingly been used for epidemiology and predictive modeling, mainly due to the effectiveness of modeling approaches on large datasets. Despite these advances, access to medical data to improve patient care remains a significant challenge: while the reasons for limited sharing of medical data are multifaceted, concerns about patient privacy are highlighted as one of the most significant issues. Uniquely, the Medical Information Mart for Intensive Care (MIMIC, Goldberger et al. 2000) database adopted a permissive access scheme which allowed for broad reuse of the data, becoming the backbone in a variety of studies ranging from assessment of treatment efficacy in well defined cohorts to prediction of key patient outcomes such as mortality, while maintaining reproducibility.

However, this type of data is heterogeneous (categorical, binary, continuous), irregularly sampled, and frequently with missing information (Singh, Sato, and Ohkuma 2021; Ren et al. 2024). This structure poses modeling challenges for most of the classic machine learning approaches, such as Logistic Regression or Random Forests, that may struggle with complex temporal and cross-modal dependencies. To alleviate the need for feature engineering, which is currently a key aspect in tabular data learning methods, and, perhaps most importantly, to allow representation and multitask learning, which enables many valuable application scenarios, Deep Learning methods, such as Recurrent Neural Networks (RNNs), Long Short Term Memory (LSTMs) and Attention-based Transformers, have improved data modeling in the EHR; however, they still treat each hospital encounter in isolation, overlooking cohort-level and longitudinal dependencies (Siebra, Kurpicz-Briki, and Wac 2024).

A solution to this problem is offered by Graph Neural Networks (GNNs) that are capable of encoding relations among admissions, patients, and clinical concepts by representing each admission as a node and encoding clinically meaningful relationships as edges. Early works built homogeneous patient-centric graphs via simple similarity metrics, while recent efforts have introduced heterogeneous graphs incorporating diagnoses, medications and procedures as separate node types. Additionally, a recurring pattern in the latest works is the use of a two-stage pipeline, where an upstream sequence or tabular model (e.g., LSTM for vital data or a Transformer for admission features) estimates intermediate clinical states or latent embeddings. Then, these outputs are used to construct downstream the graph inputs (nodes, node attributes, or edge weights), effectively denoising raw EHR features and injecting domain structure into the graph (Tong et al. 2021). However, existing methods generally do not fully exploit these intermediate embeddings directly as node features or utilize multiple edge relationships within the same patient graph.

In this work, these gaps are addressed by introducing a novel and optimized two-stage architecture (see Fig. 1.1). This approach uses a Transformer-based Clinical State Estimator to predict from admission features (gender, age, admission type, admission location, and six laboratory test results obtained in the first 24 hours since admission) both the probability of an ICU transfer and the expected delay until that transfer, but also to provide the latent embeddings that directly feed into the nodes of a heterogeneous admission-centric graph. Each node represents a distinct hospital

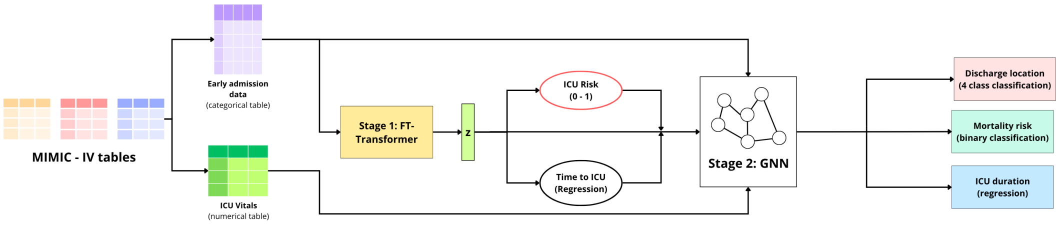

Figure 1.1. Pipeline Architecture: admission and vitals dataset aggregation; admission features are tokenized and fed to the Transformer to produce a latent representation (z) and to output ICU risk and delay; these (plus the Vitals for ICU patients) are used to enrich the graph that is processed by the GNN to predict mortality risk, ICU duration and discharge location.

admission, enriched with clinical predictions and, when available and applicable, the ICU vital signs. The nodes are connected through three complementary edge families: feature-space proximity, prospective transfer risk, and temporal continuity across a patient's multiple admissions. Finally, a hybrid GCN (Kipf et al. 2016) / R-GCN (Schlichtkrull et al. 2018) + GATv2 (Velickovic et al. 2017) + GraphSAGE (Hamilton, Ying, and Leskovec 2017) model with Jumping Knowledge (Xu et al. 2018) aggregation performs mortality risk, ICU length-of-stay, and discharge location prediction in a single forward pass.

By combining tabular transformer embeddings with a relational graph model, this pipeline aims to capture both low-level feature interactions (e.g., between gender, age, and lab tests) and high-level cohort relationships (e.g., patients with similar early states, those at risk of ICU transfer, and a patient's own longitudinal history). Together, these components demonstrate a modular and extensible framework for using tabular and graph deep learning techniques on real-world EHR data to improve clinical outcome predictions. This framework is evaluated against robust baseline models from recent literature on the MIMIC-IV (Johnson et al. 2023) database, achieving state of the art predictive performance while operating in real time on a single GPU. The results obtained demonstrate that combining Transformer embeddings directly with heterogeneous GNNs provides an effective, efficient, and clinically actionable solution for early detection of clinical deterioration and more timely ICU transfers, with significant potential to improve patient outcomes.

Chapter 2

State of the Art

2.1 Transformer Models on Tabular EHR

Attention-based architectures, such as FT-Transformer (Feature Tokenizer + Transformer, Gorishniy et al. 2022), a simple adaptation of the Transformer architecture for the tabular domain, and TabNet (Arik et al 2021), that uses sequential attention mechanisms to choose which features to reason from at each decision step, learn efficiently feature interactions, without extensive pre-processing, as the learning capacity is used for the most relevant features.

In healthcare, Transformers have been used for time-series vitals forecasting and early risk stratification from static admission data, but they ignore inter-patient relations. An et al. (2022) presented a time-aware Transformer–based Hierarchical Attention Network designed to model the irregular timing of multivariate ICU data (e.g. lab tests and vital signs) and to fuse heterogeneous modalities through a two-stage attention scheme. The Transformer learns personalized temporal decay functions for each event, while the hierarchical attention mechanism aggregates interactions into a unified patient representation.

Darabi et al. (2020) introduced TAPER, a time-aware patient EHR representation framework that first uses a Transformer encoder (with causal masking and sinusoidal positional encodings) to embed structured codes and then a BioBERT (initialized BERT plus bidirectional GRU autoencoder) to summarize clinical text into unified visit vectors.

2.2 GNNs in Clinical Predictions

GNNs have recently emerged as a powerful tool for clinical prediction tasks by explicitly modeling the relational structure inherent to medical data.

Early works adapted Graph Convolutional Networks to homogeneous patient similarity graphs, where nodes represent individual patients and edges encode similarity in demographics, comorbidity profiles or treatment histories (Boll et al. 2023; Maroudis et al. 2024; Rocheteau et al. 2021).

Subsequent work introduced heterogeneous graphs and more sophisticated architectures, characterized by multiple node and edge types, such as patients, lab tests, medications and diagnoses, linked by different relations, allowing models to jointly learn across modalities and relation semantics and to produce robust early warning systems for conditions such as sepsis and acute kidney injury (Wang and Li 2025; Daphne et al. 2025; Liu et al. 2022). In all these studies, the main advantage of GNNs lies in their ability to learn from both individual features and their complex inter-dependencies, which is crucial for accurate and clinically actionable predictions.

2.3 Two-Stage Pipelines

Two-stage architectures decompose complex prediction tasks into a coarse 'proposal' or representation phase, followed by a finer-grained refinement and decision phase, and have proven effective in many domains. Applied first to recommender systems, retrieval-then-ranking pipelines (e.g., Covington et al. 2016) use a lightweight, approximate Nearest-Neighbor model to shortlist items before a more expressive deep ranking model refines the final recommendations.

Two-stage frameworks have also been adopted in tabular and clinical settings: for example, an autoencoder or feature-tokenizer backbone produces low-dimensional patient embeddings, which are then passed to a specialized sequence model or GNN for the final predictions (Wu et al. 2020; Li et al.). Across these contexts, the two-stage paradigm balances efficiency and expressivity by allocating computational resources where they are most needed: broad-scope candidate identification followed by targeted high-capacity inference.

2.4 Multi-task Pipelines

Joint multi-task learning frameworks in clinical prediction use shared representations to improve performance across related outcomes, regularizing learning. Harutyunyan et al. (2017) first demonstrated on the MIMIC-III benchmark that RNN-based multitask models with deep supervision outperform single-task baselines on mortality, length-of-stay forecasting, physiologic decline detection, and phenotype classification.

Rajkomar et al. (2018) followed this approach by training deep sequence models on raw EHR data to simultaneously predict in-hospital mortality, 30-day readmission, prolonged length-of-stay and final discharge diagnoses.

More recently, Shickel et al. (2021) introduced a Flexible Multimodal Transformer that jointly predicts several ICU outcomes, including various readmission windows and mortality horizons, within one end-to-end architecture, showcasing state of the art accuracy and the ability to incorporate diverse data modalities. By exploiting common signals across tasks, these joint multi-task pipelines yield more robust, data-efficient, and interpretable clinical decision support systems.

Chapter 3

Dataset

3.1 MIMIC-IV

The MIMIC-IV (Medical Information Mart for Intensive Care) is a large de-identified clinical database drawn from patients admitted to the emergency department or an Intensive Care Unit at the Beth Israel Deaconess Medical Center in Boston, MA. It contains data from 2008 to 2022 for more than 65,000 patients admitted to an ICU and over 200,000 patients admitted to the emergency department, reflecting present-day clinical practice with ICD-10 diagnosis codes, electronically charted medication administration, and other contemporary workflows. In detail, this database is divided into two modules: ’hosp’ and ’icu’: data in the ’hosp’ module is sourced from the hospital wide EHR, while data in the ’icu’ module is sourced from the in-ICU clinical information system; this modular approach to data organization highlights data provenance and facilitates both individual and combined use of disparate data sources (Fig. 3.1).

Patient identifiers, as stipulated by HIPAA, have been removed and replaced using a random cipher, resulting in de-identified integer identifiers for patients, hospitalizations, and ICU stays.

3.2 Aggregation and Preprocessing

For preprocessing the data, specific columns were extracted and their values were aggregated to better fit into a neural network.

3.2.1 Administrative and Demographic Table

A first table is built from both the ’hosp’ and ’icu’ modules to capture the demographic and administrative details of each admission: hospital admission ID,

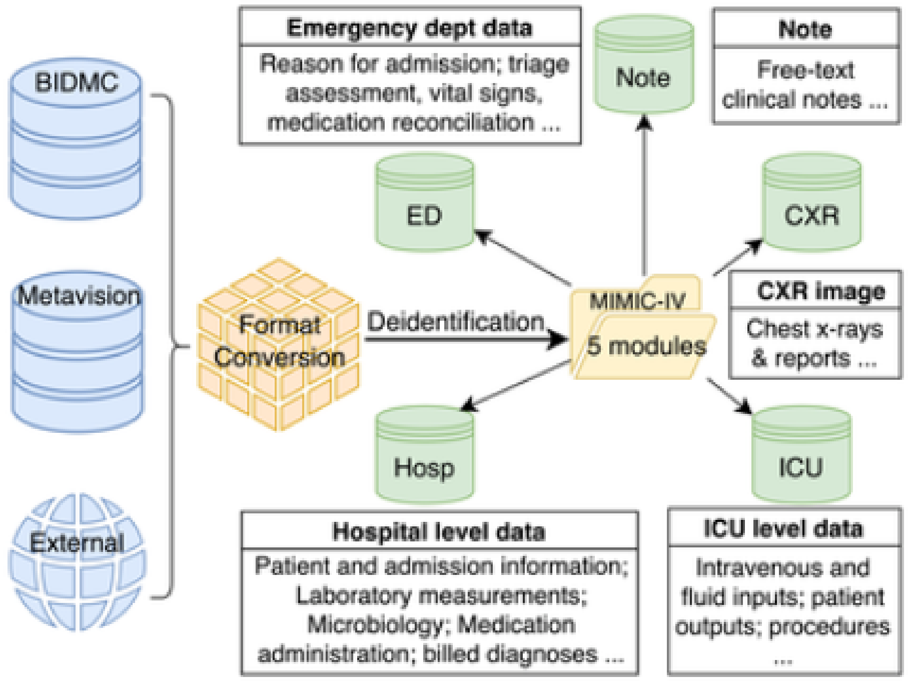
BIDMC
Metavision
External
Format Conversion
Deidentification
MIMIC-IV
5 modules
Emergency dept data
Reason for admission; triage assessment, vital signs, medication reconciliation ...
Note
Note
Free-text clinical notes ...
ED
CXR
CXR image
Chest x-rays & reports ...
Hosp
ICU
Hospital level data
Patient and admission information; Laboratory measurements; Microbiology; Medication administration; billed diagnoses ...
ICU level data
Intravenous and fluid inputs; patient outputs; procedures ...

Figure 3.1. MIMIC-IV built by merging the various data sources and structured in five different modules, each with its own specific tables.

patient ID, gender, age, admission logistics (specifying the admission type: elective, emergency or observation), source location (where the patient came from), discharge disposition (where the patient was sent after leaving the hospital: home, deceased, transfer, other); a variable is added to identify whether the patient, during that specific admission, went to an ICU or not. The age of the patients was discretized into 8 ordered age groups (0-20,21-30,31-40.41-50,51-60,61-70,71-80,81-99).

3.2.2 Early Laboratory Results in the First 24 Hours

For each lab test (bicarbonate, creatinine, glucose, ast, bilirubin, hematocrit), a categorical flag was introduced: if *any* measurement of that test in the first 24 hours after admission was marked as ’abnormal’, the flag was set to ’abnormal’; otherwise, if there were all ’normal’ or missing values, the flag was set to ’normal’. Then, these results were transformed into binary indicators.

3.2.3 Vital Signs (for ICU-admissions only)

In addition, a separate table was created to summarize the vital signs measurements recorded for each admission during stays in the ICU. For each ICU-admission

ID, the average, the minimum, and the maximum values were calculated for all available ICU chart events (Systolic Blood Pressure, Heart Rate, Respiratory Rate, Body Temperature, Peripheral Oxygen Saturation (SpO2), Point-of-Care Glucose) and z-score normalized, resulting in a distribution with average 0 and standard deviation 1, to balance the absence of some or all of these statistics for many admissions. The purpose of this table was to condense hundreds of individual measurements into a few clinically meaningful metrics for each admission, balancing predictive accuracy, interpretability, and computational and runtime costs.

3.2.4 ICU Timing Intervals

For each admission, the time from hospital admission to the first ICU entry (in hours) and the duration of the ICU stay (in hours) were computed: these two values are used as model target data and have been min-max normalized to stabilize training and improve convergence.

3.3 Dataset Balancing

To address the class imbalance problem in the `was_in_icu` label, all the ICU admissions were extracted (85 242 rows) from the dataset (546 028 admissions) and concatenated to 127 863 randomly selected non-ICU admissions, producing a final dataset of 213 105 rows, where the ICU cases represent the 40% of the total (still realistic but more balanced).

A final cleaning step removed extreme outliers: any admission whose `icu duration` was in the top 1% (from longest to shortest) was flagged as a ’big outlier’ and removed, resulting in a dataset that excludes extreme length-of-stay values (Table 3.1).

A summary of the variables contained in both our custom tables is shown in Table 3.2 and 3.3.

Table 3.1. Summary of Dataset Balancing and Outlier Removal

Total admissions	546 028
ICU admissions (`was_in_icu` = 1)	85 242 (15.6%)
ICU ratio after balancing	40%
Final dataset size (after balancing)	213 105
Rows removed (top 1% `icu_duration`)	2 131
Mean `icu_duration` before removal	84.20
Max `icu_duration` before removal	3 832.00
Mean `icu_duration` after removal	69.45
Max `icu_duration` after removal	421.66
ICU ratio after outlier removal	39.39%

Table 3.2. Dataset Columns: Attributes, Data Types and MIMIC-IV Sources

Attribute	**Data Type**	**Source MIMIC-IV Table**
hadm_id	Identifier (integer)	admissions
subject_id	Identifier (integer)	patients
gender	Binary (M/F)	patients
age_group	Categorical (8 bins)	patients (anchor_age)
year	Integer (admission year)	patients (anchor_year_group)
hosp_mortality	Binary (0/1)	admissions (deathtime)
adm_type	Multiclass (4 categories)	admissions
adm_loc	Multiclass (4 categories)	admissions
disc_loc	Multiclass (4 categories)	admissions
bicarbonate_test	Binary (normal/abnormal)	labevents (itemid=50868)
creatinine_test	Binary (normal/abnormal)	labevents (itemid=50882,50912)
glucose_test	Binary (normal/abnormal)	labevents (itemid=50931,50893)
ast_test	Binary (normal/abnormal)	labevents (itemid=50862)
bilirubin_test	Binary (normal/abnormal)	labevents (itemid=50878,50885)
hematocrit_test	Binary (normal/abnormal)	labevents (itemid=50822,50810)
was_in_icu	Binary (0/1)	icustays
mean_sysbp	Continuous (mmHg)	chartevents (itemid=220045)
min_sysbp	Continuous (mmHg)	chartevents (itemid=220045)
max_sysbp	Continuous (mmHg)	chartevents (itemid=220045)
mean_hr	Continuous (beats/min)	chartevents (itemid=220179)
min_hr	Continuous (beats/min)	chartevents (itemid=220179)
max_hr	Continuous (beats/min)	chartevents (itemid=220179)
mean_rr	Continuous (breaths/min)	chartevents (itemid=220180)
min_rr	Continuous (breaths/min)	chartevents (itemid=220180)
max_rr	Continuous (breaths/min)	chartevents (itemid=220180)
mean_temp	Continuous (°C)	chartevents (itemid=220210)
min_temp	Continuous (°C)	chartevents (itemid=220210)
max_temp	Continuous (°C)	chartevents (itemid=220210)
mean_spo2	Continuous (%)	chartevents (itemid=220277)
min_spo2	Continuous (%)	chartevents (itemid=220277)
max_spo2	Continuous (%)	chartevents (itemid=220277)
mean_glucose	Continuous (mg/dL)	chartevents (itemid=223761)
min_glucose	Continuous (mg/dL)	chartevents (itemid=223761)
max_glucose	Continuous (mg/dL)	chartevents (itemid=223761)
hosp_to_icu_1	Continuous (hours)	icustays & admissions
icu_duration_1	Continuous (hours)	icustays

Table 3.3. Distribution of the categorical attributes of the dataset

Attribute	Label	Count	%
Gender	0	106,413	50.4%
	1	104,561	49.6%
In-hospital mortality	0	199,753	94.7%
	1	11,221	5.3%
Admission type	0	29,488	14.0%
	1	138,289	65.6%
	2	43,197	20.5%
Admission location	0	85,277	40.4%
	1	2,456	1.2%
	2	83,728	39.7%
	3	39,513	18.7%
Discharge location	0	14,521	6.9%
	1	145,706	69.1%
	2	1,644	0.8%
	3	49,103	23.3%
Was in ICU	0	127,863	60.6%
	1	83,111	39.4%
Bicarbonate test	0	23,668	11.2%
	1	187,306	88.8%
Creatinine test	0	79,595	37.7%
	1	131,379	62.3%
Glucose test	0	133,936	63.5%
	1	77,038	36.5%
AST test	0	23,745	11.3%
	1	187,229	88.7%
Bilirubin test	0	31,344	14.9%
	1	179,630	85.1%
Hematocrit test	0	8,429	4.0%
	1	202,545	96.0%
Age group	0	2,695	1.3%
	1	15,727	7.5%
	2	18,335	8.7%
	3	24,254	11.5%
	4	39,162	18.6%
	5	45,090	21.4%
	6	36,549	17.3%
	7	29,162	13.8%

Chapter 4

Methodology

4.1 Stage 1

In recent years, many different approaches have been adopted to handle EHR data, which commonly include many discrete low-cardinality variables (e.g., gender, age group, admission type, location codes, binary lab flags). In particular, in this work the performances of Logistic Regression (for ICU risk classification only), Linear Regression (for time to ICU regression only), Random Forests, Multi-Layer Perceptron (MLP) and Feature-Tokenizer-Transformer (FT-Transformer) are compared. Tables 5.1 and 5.2 (shown in Chapter 5) summarize the results obtained in the experiments: the transition from linear to nonlinear and then to DL approaches produced incremental gains, the addition of multitask supervision consistently improved both classification and regression and, finally, the FT-Transformer (Figure 4.1) produced the greatest benefits, allowing also multi-task learning and the encoding of a latent representation.

4.1.1 Multi-task Transformer-Based Clinical State Estimator

The Feature Tokenizer module learns to project the categorical features in input $\mathbf{x}$ into embeddings $\mathbf{T} \in R^{k \times d}$, where k is the number of features and d their dimensionality. The embedding for a given feature x_j is computed as follows:

$$\mathbf{T}_j = \mathbf{b}_j + f_j(x_j), \quad f_j : \mathcal{X}_j \to R^d \tag{4.1}$$

For categorical features, $f_j^{(\text{cat})}$ is implemented as a lookup table $\mathbf{W}_j^{(\text{cat})} \in R^{S_j \times d}$:

$$\mathbf{T}_j^{(\text{cat})} = \mathbf{b}_j^{(\text{cat})} + \mathbf{e}_j^{\top} \mathbf{W}_j^{(\text{cat})} \in R^d \tag{4.2}$$

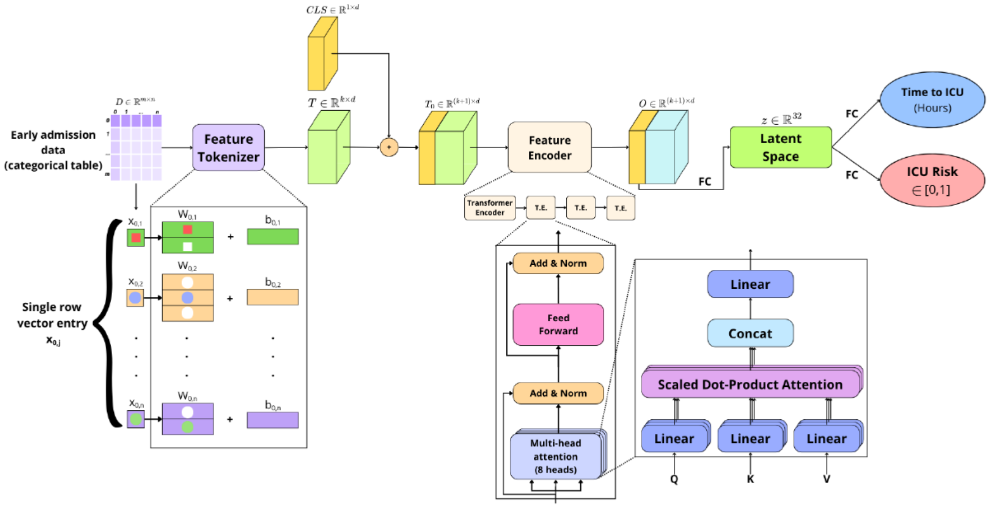

Figure 4.1. Feature-Tokenizer Transormer Architecture: admission features are tokenized and fed to the Transformer; it produces a latent representation (z) and outputs ICU risk and delay through its Multi-Head Attention mechanism and a final Fully Connected (FC) block.

where $\mathbf{e}_j$ is a one-hot vector for the corresponding categorical feature. The complete embedding $\mathbf{T}$ is constructed as:

$$\mathbf{T} = \text{stack}\left[\mathbf{T}_1^{(\text{cat})}, \ldots, \mathbf{T}_{k_\text{cat}}^{(\text{cat})}\right] \in R^{k\times d} \tag{4.3}$$

Subsequently, a learnable classification token [CLS] (Devlin et al. 2019) is concatenated to the input embeddings T; the model uses it during training, to optimize how to summarize the feature embeddings into a single vector. This has proven effective in NLP and, more recently, in tabular tasks (Gorishniy et al. 2022).

$$\mathbf{T}_0 = \text{stack}\left[[\text{CLS}], \mathbf{T}\right], \quad \mathbf{T}_i = F_i(\mathbf{T}_{i-1}) \tag{4.4}$$

By stacking a small Transformer encoder (four layers, eight heads), the network can learn higher-order interactions among all categorical tokens. The intuition is that certain combinations of discrete attributes can jointly signal ICU-risk in ways that linear or shallow MLPs cannot easily capture. The Transformer self-attention mechanism can flexibly model interactions between all categorical features without explicitly hand-crafting cross-features. Attention layers can attend globally and thus adapt better to heterogeneous categories compared to a standard MLP on concatenated one-hot inputs. Finally, the [CLS] token is passed to an MLP block, followed by Layer Normalization and Dropout (0.3), which help mitigate overfitting (particularly important when many categories are imbalanced, such as certain admission types or rare lab flags). This MLP simultaneously outputs a logit for ICU-risk (binary classification) and a scalar prediction of "delay until ICU transfer" (regression). This multi-task formulation encourages the shared latent representation ($z \in \mathbb{R}^{32}$), used as node feature in the second stage, to encode both information. In healthcare settings, predicting both "if" and "when" a patient deteriorates can provide more helpful clinical decision support than a single binary risk score. The loss used to train this network is the sum of the two individual ones:

$$\hat{y}_i^{(\text{Risk})} = \sigma\Big(\mathbf{w}_{\text{Risk}}^{\top}\mathbf{h}_i + b_{\text{Risk}}\Big) \tag{4.5}$$

$$\hat{y}_i^{(\text{Time})} = \mathbf{w}_{\text{Time}}^{\top}\mathbf{h}_i + b_{\text{Time}} \tag{4.6}$$

$$\mathcal{L}_{\text{Risk}} = -\Big[\, y_i \log \hat{y}_i^{(\text{Risk})} + (1-y_i)\log\Big(1-\hat{y}_i^{(\text{Risk})}\Big)\Big] \tag{4.7}$$

$$\mathcal{L}_{\text{Time}} = \frac{1}{N}\sum_i \left(\hat{y}_i^{(\text{Time})} - y_i^{(\text{Time})}\right)^2 \tag{4.8}$$

The overall loss is defined as the sum of the two components:

$$\mathcal{L} = \mathcal{L}_{\text{Risk}} + \mathcal{L}_{\text{Time}}. \tag{4.9}$$

4.2 Stage 2

4.2.1 Multi-edge Graph Construction

A graph is a fundamental mathematical structure that models relationships between entities. Formally, a graph is defined as an ordered pair $G = (V, E)$, where $V = \{v_1, v_2, \ldots, v_N\}$ is a finite set of nodes (or vertices) representing entities and $E \subseteq V \times V$ is a set of edges that represent pairwise relationships or interactions between them. An edge $e_{ij} = (v_i, v_j) \in E$ indicates a connection between nodes v_i and v_j. Graphs can be undirected, where $(v_i, v_j) = (v_j, v_i)$, or directed, where edges have an orientation. In weighted graphs, a function $w : E \to \mathbb{R}$ assigns a weight to each edge, encoding the strength or importance of the connection. A graph can also be represented algebraically through an adjacency matrix $A \in \mathbb{R}^{N\times N}$, where each entry A_{ij} denotes the weight of the edge between v_i and v_j (or 0 if no edge exists). This formalism provides the foundation for many algorithms and models in network analysis and Graph Neural Networks (GNNs).

In this work, a multi-edge admission-centric graph (Figure 4.2) is built to represent each hospital admission as a node, whose features are composed of admission data, latent representation (z) extracted from Stage-1, ICU Risk, Time to ICU, and Vitals

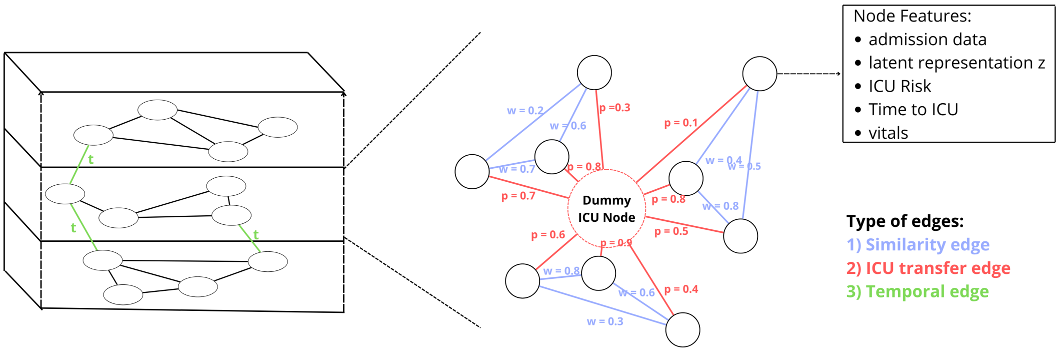

Figure 4.2. Admission-centric graph construction: each node represents an admission, characterized by admission data, latent representation, ICU risk, time to ICU and ICU Vitals; nodes are linked to each other through weighted similarity edges, ICU transfer edges (to a dummy node) and within-subject edges.

(if the admission was in ICU). Nodes are connected through three edge families: the feature-space similarity edges are drawing according to the K-Nearest Neighbors (k=5) algorithm applied to the joint space of the node features and connect the admission nodes between each other, enforcing the idea that patients with similar early-stage features share important information.

The ICU transfer risk edges connect each admission node to a single Dummy Node and are weighted by the ICU Risk, calculated from Stage, implementing a star topology that aggregates global ICU knowledge. The purpose of the dummy node is to pool information about all admissions, handling scalability and oversmoothing issues in graph structure learning (Liu et al. 2022).

Finally, the temporal edges link all admissions of the same patient during the years, capturing longitudinal progression. In detail, a temporal link starting in node_i and arriving in node_j is weighted by the following formula:

$$w_{ij} = \frac{1}{1 + |\text{year}_i - \text{year}_j|} \tag{4.10}$$

where $\text{year}_i < \text{year}_j$. By doing so, the Stage 2 network can use the past history when predicting current outcomes (Gupta et al. 2025).

4.2.2 Multi-layer GNN

The generated graph is then processed by a GNN pipeline that consists of a GCN / R-GCN, a GATv2, and a GraphSAGE, followed by separated linear heads for each task. This multi-layer architecture combines different aggregation and message-passing mechanisms to capture complementary relational patterns across hospital stays.

A first GCNConv layer incorporates the local neighborhood structure and projects the raw node features into a hidden representation. Let $H^{(0)} \in R^{N \times d_0}$ be the matrix of initial node features, where N is the number of nodes and d_0 their dimensionality, and $A \in R^{N \times N}$ the weighted adjacency matrix; this layer applies:

$$H^{(1)} = \sigma\Big(\tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}H^{(0)}W^{(0)}\Big) \tag{4.11}$$

where $\tilde{A} = A + I$, $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$, $\sigma(\cdot)$ is the ReLU activation function, and $W^{(0)} \in R^{d_0 \times d_1}$. This architecture is simple and deterministic, accounts for edge weights during aggregation and is efficient on CPU with sparse matrices, but it does; however, it does not capture edge heterogeneity.

To address this problem, an alternative design uses an R-GCNConv (Schlichtkrull et al. 2018) as first layer, to model heterogeneous edge relations explicitly. It uses relation-specific transformation matrices $\{W_r\}$ and aggregates messages across different edge types as:

$$h_i^{(1_r)} = \sigma\left(\sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{N}_i^r} \frac{1}{c_{i,r}} W_r h_j^{(1)} + W_0 h_i^{(1)}\right) \tag{4.12}$$

where $\mathcal{R}$ denotes the set of relations and $c_{i,r}$ is a normalization constant. This allows the network to learn different propagation patterns for different clinical interactions while controlling parameter growth through basis decomposition.

Subsequently, a second GATv2Conv layer (Brody et al. 2022) applies adaptive multi-head attention to dynamically weigh the relative importance of neighboring nodes. For each node, attention coefficients are computed between neighbors and aggregated through a weighted average; for each edge (i, j):

$$e_{ij} = \mathrm{LeakyReLU}\Big(\mathbf{a}^{\top}\big[\mathbf{W}^{(1)}\mathbf{h}_i^{(1)} \,\|\, \mathbf{W}^{(1)}\mathbf{h}_j^{(1)}\big]\Big), \tag{4.13a}$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum\limits_{k \in \mathcal{N}(i)} \exp(e_{ik})}, \tag{4.13b}$$

$$\mathbf{h}_i^{(2)} = \sigma\Big(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}\, \mathbf{W}^{(1)}\mathbf{h}_j^{(1)}\Big). \tag{4.13c}$$

In this way, this layer learns a different importance for each neighbor, improving expressiveness compared to a GCN at the cost of a bit higher computational consumption (attention on each edge).

Then, a third and last SAGEConv layer aggregates information from multi-hop neighbors and learns node-specific aggregation functions (e.g. *mean*, *max-pooling*, *LSTM*), enriching the contextual representation over neighbors for each node.

$$\mathbf{m}_i = \mathrm{AGG}\big(\{\mathbf{h}_j^{(2)} : j \in \mathcal{N}(i)\}\big) \tag{4.14a}$$

$$\mathbf{h}_i^{(3)} = \sigma\big(\mathbf{W}^{(2)}\,[\mathbf{h}_i^{(2)} \,\|\, \mathbf{m}_i]\big) \tag{4.14b}$$

It is *inductive*, generating embeddings for new nodes using only their features, without requiring a global re-computation (scalable on large graphs through neighbor sampling). The complexity per layer is $\mathcal{O}(|E|)$, but is still efficient.

Finally, the embeddings from all layers $h_i^{(j)} \in R^{d_j}$ are concatenated using the Jumping Knowledge mechanism (mode=`cat`) to balance early and late feature contributions:

$$\mathbf{h}_i^{\mathrm{JK}} = \Big[\mathbf{h}_i^{(0)} \,\|\, \mathbf{h}_i^{(1)} \,\|\, \dots \,\|\, \mathbf{h}_i^{(3)}\Big] \tag{4.15}$$

This “skip connection” style aggregation was introduced by Xu et al. (2018), founding that combining multiple types of layer (in particular GCN, GAT and SAGE) leads to a more stable training and improves accuracy on sparse or relational healthcare graphs.

Then, the resulting embedding is fed to three separate linear heads: `Mortality Risk`, `Length of ICU Stay`, and `Discharge Location`.

$$\hat{y}_i^{(\mathrm{mort})} = \sigma\big(\mathbf{w}_{\mathrm{mort}}^{\top}\mathbf{h}_i^{\mathrm{JK}} + b_{\mathrm{mort}}\big), \tag{4.16a}$$

$$\hat{y}_i^{(\mathrm{hours})} = \mathbf{w}_{\mathrm{hours}}^{\top}\mathbf{h}_i^{\mathrm{JK}} + b_{\mathrm{hours}}, \tag{4.16b}$$

$$\hat{y}_i^{(\mathrm{disc})} = \mathrm{softmax}\big(\mathbf{W}_{\mathrm{disc}}\,\mathbf{h}_i^{\mathrm{JK}} + \mathbf{b}_{\mathrm{disc}}\big), \tag{4.16c}$$

$$\mathcal{L}_{\mathrm{mort}} = -\big[y_i \log \hat{y}_i^{(\mathrm{mort})} + (1 - y_i)\log\big(1 - \hat{y}_i^{(\mathrm{mort})}\big)\big], \tag{4.16d}$$

$$\mathcal{L}_{\mathrm{hours}} = \frac{1}{N}\sum_i \big(\hat{y}_i^{(\mathrm{hours})} - y_i^{(\mathrm{hours})}\big)^2, \tag{4.16e}$$

$$\mathcal{L}_{\mathrm{disc}} = -\sum_{c=1}^{4} y_{i,c}^{(\mathrm{disc})}\,\log \hat{y}_{i,c}^{(\mathrm{disc})}. \tag{4.16f}$$

The overall loss is a (weighted) sum of the three individual ones:

$$\mathcal{L} = \lambda_{\mathrm{mort}}\mathcal{L}_{\mathrm{mort}} + \lambda_{\mathrm{hours}}\mathcal{L}_{\mathrm{hours}} + \lambda_{\mathrm{disc}}\mathcal{L}_{\mathrm{disc}}. \tag{4.17}$$

This joint training strategy encourages the network to produce useful embeddings for all relevant clinical tasks at once, allowing to learn common representations useful for multiple tasks and to reduce the total number of parameters.

Furthermore, during training, Dynamic Edge Weighting via MLP is applied to recalculate weights based on the concatenated hidden states of source/destination nodes and the previous edge weight: in practice, the GNN learns to emphasize or de-emphasize certain connections dynamically.

$$\mathbf{e}_{ij} = [\, \mathbf{h}_i^{(l)} \parallel \mathbf{h}_j^{(l)} \parallel w_{ij}^{(0)} \,], \tag{4.18a}$$

$$w_{ij}^{(l)} = \mathrm{MLP}_{\mathrm{edge}}(\mathbf{e}_{ij}), \tag{4.18b}$$

$$\mathbf{m}_i^{(l+1)} = \sum_{j \in \mathcal{N}(i)} w_{ij}^{(l)} \cdot \mathbf{h}_j^{(l)}. \tag{4.18c}$$

Edges are also randomly dropped during training (with `dropedge_rate`$= 0.2$), to avoid over-reliance on a few strong connections and to reduce overfitting. DropEdge Regularization was introduced by Rong et al. (2020) and has since become a common practice in graph learning.

Chapter 5

Results

5.1 Stage 1 Results Comparison

Tables 5.1 and 5.2 summarize the performance of the five different modeling approaches implemented for the two ICU-prediction tasks. Overall, we can see that traditional methods, such as LogReg and Random Forests, yield reasonable performances on ICU risk prediction (AUC 0.75, Acc 0.70) and timing errors around 45 h (MAE); we can also notice that the usage of an MLP, in particular with the addition of multi-task learning, provides modest gains (AUC 0.76, Acc 0.71, MAE 40 h). Finally, the Transformer-based multi-task model further improves discrimination (AUC 0.77, confusion matrix in Table 5.3) and timing accuracy (MAE 35 h), suggesting that its richer feature interactions can better capture both the classification and temporal aspects of ICU risk .

Compared to recent studies performed on the MIMIC-IV dataset, there are no benchmarks on ICU risk and delay, but previous and similar works on different datasets reported an ICU-risk AUC of around 0.75-0.80 using LSTM-based estimators or models using embeddings of categorical features with attention.

5.2 Stage 2 Layer Ablation Study

Table 5.4 summarizes the impact of different GNN configurations in the Stage 2 architecture. The baseline combination GCN + GAT + SAGE already achieves strong overall performance in both classification and regression tasks. The replacement of the GCN layer with a Relational Graph Convolution further improves the results in all metrics, with the entire R-GCN + GAT + SAGE model achieving the best overall performance (AUC 0.909, RMSE 54.6). This indicates that explicitly modeling edge types and relational dependencies enhances representation learning and stability

Table 5.1. ICU Risk prediction: Performance Summary across various classical ML and DL approaches

Model	**AUC**	**Accuracy**
Logistic Regression	0.751	0.701
Random Forest	0.756	0.707
MLP (single-task)	0.757	0.707
MLP (multi-task)	0.762	**0.710**
FT-Transformer (multi-task)	**0.770**	**0.710**

Table 5.2. Time to ICU Regression: Performance Summary across various classical ML and DL approaches

Model	**RMSE [h]**	**MAE [h]**
Linear Regression	**90**	45
Random Forest	92	45
MLP (single-task)	92	45
MLP (multi-task)	**90**	40
FT-Transformer (multi-task)	92	**35**

during message passing.

Monolayer variants show slightly degraded results, confirming that multi-layer aggregation provides complementary information. In particular, the R-GCN variants produce better discrimination in long ICU-stays (AUC 0.86 for ICU duration $>$ 3 days), suggesting improved temporal and relational reasoning across patient trajectories.

5.3 Stage 2 Results Comparison

Table 5.5 summarizes performance across configurations. The first two columns from the left are the results obtained with/without outliers in the case in which the ICU Vitals are assigned with an ICU Risk $>$ 0.5 from Stage 1; the third column from the left contains the results obtained when Vitals are assigned to all ICU admissions using the Ground Truth (GT) of the label 'was in ICU', removing the potential noise from Stage 1: this can be considered reasonable, knowing that vitals are available only if a patient has been in an ICU department and this fact, in a real-word scenario, may depend just in a first instance on the prediction from Stage

Table 5.3. Confusion Matrix for ICU Risk Prediction (>0.5)

	Predicted		
Actual	**ICU**	**Not ICU**	**Total**
ICU	3 730	4 564	8 294
Not ICU	1 584	11 220	12 804
Total	5 314	15 784	21 098

Table 5.4. Ablation study on different GNN configurations for Stage 2. Metrics include AUC and Accuracy for Mortality Risk, RMSE and MAE for ICU Duration, Accuracy for Discharge Location, and AUC for ICU duration $>$ 3 days and $>$ 7 days.

Model	**AUC(Mort)**	**Acc(Mort)**	**RMSE(h)**	**MAE(h)**	**Acc(DiscLoc)**	**AUC(>3d)**	**AUC(>7d)**
Monolayer GCN	0.894	0.961	57.5	37.6	0.740	0.841	0.876
Monolayer GAT	0.902	0.962	55.7	37.0	0.741	0.848	0.888
Monolayer SAGE	0.904	0.961	57.0	37.1	0.739	0.843	0.875
Monolayer R-GCN	0.907	0.962	54.7	35.6	0.741	**0.859**	0.898
GCN + GAT	0.896	0.961	56.9	37.9	0.739	0.842	0.883
GCN + SAGE	0.898	0.962	56.4	35.5	**0.742**	0.848	0.883
GAT + SAGE	0.905	0.961	56.7	36.6	0.738	0.845	0.879
GCN + GAT + SAGE	0.903	0.962	55.6	36.7	0.740	0.850	0.890
R-GCN + GAT + SAGE	**0.909**	**0.963**	**54.6**	**35.4**	0.740	0.856	**0.900**

1 and more likely on the actual clinical condition of the patient. We can consider this solution, and the corresponding results, as an "upper bound" for Stage 2.

Compared to the performance without outliers removal (AUC 0.8541, Accuracy 0.9562), after this cleaning step, the AUC decreased slightly (0.840), while the accuracy remained unchanged or slightly improved (0.9575). Since approximately 2100 admissions were removed (some of which probably included patients with extremely long ICU stays and atypical clinical profiles), this slight drop in AUC is expected: the model now mostly encounters admissions with more typical outcomes. However, the accuracy remains high, as the removal of outliers has little effect on the many ”easy” cases of survival or non-survival. The impact of outlier removal is most evident in the ICU-Duration regression: RMSE was halved (from 120 to 63 hours) and MAE significantly decreased (from 72 to 42 hours).

In the last case, the graph-based mortality risk predictor achieved an AUC of 0.909 and an Accuracy of 0.963, surpassing the best publicly reported MIMIC-IV benchmarks (Table 5.6): mortality risk AUC around 0.90, obtained using a Bert Transformer for textual embeddings followed by a GNN (Daphne et al. 2025). Instead, the performance for the ICU-duration regression (MAE 35 h, RMSE 54 h) is slightly higher than the 37-40 h RMSE range reported by Rocheteau et al. (2021),

Table 5.5. Performance Summary Across Experimental Conditions: before and after outliers removal and with the assignment of ICU Vitals using the ground truth of 'was in ICU'.

Metric	Outliers	No Outliers	GT Vitals
Mortality AUC	0.854	0.840	**0.909**
Mortality Acc	0.956	0.957	**0.963**
ICU RMSE (h)	120	63	**55**
ICU MAE (h)	72	42	**35**
Discharge Acc	0.722	0.722	**0.740**

Table 5.6. Performance comparison with related works on MIMIC-III/IV for mortality risk prediction

Work	**Model**	**AUC**
Rocheteau et al. (2021)	LSTM	0.83
	LSTM+GNN	0.86
Bui et al. (2024)	XGBoost	0.87
	LSTM	0.83
van de Water et al. (2025)	GRU	0.87
Daphne et al. (2025)	Transformer+GNN	0.90
This Work	Transformer+GNN	**0.91**

using an LSTM + GNN. Finally, when considering the AUC for ICU duration >3 days and >7 days (respectively 0.84 and 0.90 with the LSTM + GNN by Maroudis et al. 2024), it achieves an AUC of 0.86 and 0.90, respectively. This indicates broadly comparable performance and demonstrates that, once true ICU admissions are known, the GNN can achieve nearly optimal predictive performance.

Figure 5.1 and tables 5.7 and 5.8 report some interesting and promising results obtained during the experiments.

5.4 Interpretability

Interpretability is crucial in clinical decision support systems to ensure that model predictions can be trusted and validated by healthcare professionals. In this two-stage pipeline, the output of the GNN combines complex interactions across categorical embeddings, latent clinical state, and relational edges (similarity, ICU

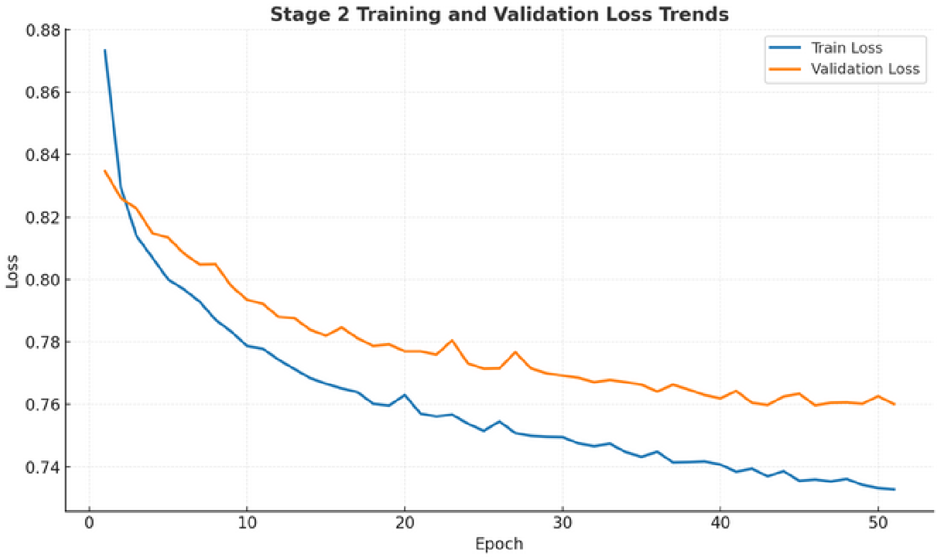

Figure 5.1. Training and validation loss across epochs for the GNN (in this case, in particular, R-GCN+GAT+SAGE). The validation loss stabilizes around epoch 40, where early stopping is applied.

Table 5.7. Confusion Matrix for Mortality Risk Prediction (>0.5)

	Predicted		
Actual	**Dead**	**Not Dead**	**Total**
Dead	498	627	1 125
Not Dead	170	19 803	19 973
Total	668	20 430	21 098

transfers, temporal history). To shed light on "why" the model predicts a high mortality risk or a long ICU-stay for a specific patient, two different approaches are implemented.

The first is based on a simple feature perturbation mechanism (for each input dimension of the selected admission node, the features are zeroed-out, observing how much the mortality risk or the ICU-duration changes; the resulting importance scores identify which inputs drive the final prediction (Table 5.9). This straightforward perturbation technique provides an intuitive and model-agnostic way to rank features by their influence, enabling clinicians to verify that critical variables (e.g. age group, specific lab flags or prior ICU risk) align with medical knowledge when the model classifies a patient as high-risk.

The second and more advanced method is based on the GNNExplainer (Ying et al.,

Table 5.8. ICU-Duration Prediction Error

Error Interval	Count	% of ICU	Cumulative %
$\leq$ 6 h	1,234	14.88%	14.88%
6–12 h	1,175	14.17%	29.05%
12–24 h	1,801	21.71%	50.76%
24–36 h	1,206	14.54%	65.30%
36–48 h	868	10.47%	75.77%
48–72 h	876	10.56%	86.33%
72–96 h	446	5.38%	91.71%
96–120 h	260	3.13%	94.84%
$\geq$ 120 h	428	5.16%	100.00%
Total	8,294	100.00%	100.00%

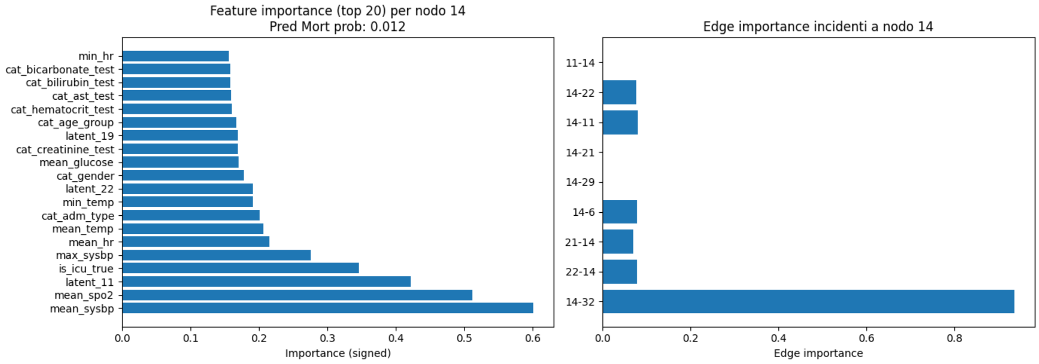

Figure 5.2. Example of Feature and Edge Importance produced by Explainer (GNNExplainerAlgo by torch geometric).

NeurIPS 2019): it identifies the most influential subgraph structure and node features that contribute to a given prediction by optimizing a mutual information objective between the full model prediction and the explanation subgraph. Specifically, it learns a soft mask over edges and features to highlight the minimal subset that preserves the model's output for a target node or graph. In addition to the previous and simpler approach, this allows visualization of clinically relevant relations, such as which node connections drive a particular risk prediction, thus improving transparency and trust in the model's decisions. This step helps establish transparency and supports clinical validation before deploying the system in practice (Fig. 5.2).

Table 5.9. Feature Importance, ordered by magnitude, for Mortality Prediction Task (True Mortality, Predicted Probability = 0.873).

Feature	Importance	Feature	Importance
min_sysbp	0.789883	max_temp	0.001556
latent_25	0.752459	max_sysbp	0.001142
age_group	0.429172	max_glucose	-0.001022
latent_10	0.396917	max_spo2	-0.000947
adm_type	0.208608	max_hr	-0.000278
mean_temp	0.116018	gender	0.000000
mean_sysbp	0.114884	adm_loc	0.000000
latent_6	-0.087140	bicarbonate_test	0.000000
min_rr	0.086561	creatinine_test	0.000000
mean_hr	-0.064819	glucose_test	0.000000
latent_28	0.060689	ast_test	0.000000
min_temp	-0.043527	bilirubin_test	0.000000
latent_23	0.042981	hematocrit_test	0.000000
time_to_icu	0.041880	latent_0	0.000000
mean_rr	-0.040183	latent_2	0.000000
min_spo2	-0.038642	latent_4	0.000000
is_icu_pred	-0.037838	latent_5	0.000000
latent_29	0.037322	latent_7	0.000000
latent_21	-0.026424	latent_9	0.000000
min_hr	0.024553	latent_11	0.000000
mean_glucose	-0.019785	latent_13	0.000000
latent_1	0.018036	latent_14	0.000000
latent_12	-0.017621	latent_15	0.000000
latent_3	0.010438	latent_16	0.000000
latent_31	0.009088	latent_17	0.000000
latent_30	0.007445	latent_18	0.000000
latent_20	-0.006707	latent_19	0.000000
latent_8	-0.006500	latent_22	0.000000
max_rr	-0.004530	latent_24	0.000000
mean_spo2	0.004234	latent_26	0.000000
min_glucose	-0.003295	latent_27	0.000000

Chapter 6

Conclusions

In this work, a modular two-stage deep learning pipeline has been introduced to combine Transformer-based embeddings with a multi-edge GNN and to predict multiple clinical outcomes from the MIMIC-IV database.

Stage 1 Transformer encoder produces from raw admission features a compact latent representation while jointly estimating ICU transfer risk and time to ICU, achieving a slight improvement over classical baselines.

Stage 2 then uses these embeddings, together with similarity, ICU-transfer, and temporal edges, within a heterogeneous admission-centric hypergraph to perform multi-task prediction of in-hospital mortality, length of stay in ICU, and discharge location. Empirical results demonstrate state-of-the-art performance on MIMIC-IV, with an AUC of 0.91 for mortality risk, MAE of 35 h for ICU stay, and discharge prediction accuracy of 0.74.

This architecture achieves an inference time of around 1 second for the prediction of Stage 1 and 3 seconds for Stage 2 on the AMD Ryzen 5 7600X, while runs in real-time on standard GPUs.

In addition, the feature importance analysis offers an expressive interpretability mechanism, allowing clinicians to trace which factors most strongly influence each prediction, helping the integration into real-world decision support workflows.

In the future, in a real-word scenario, an additional module, such as a Bert Transformer for NLP, could be used to incorporate texts of diagnoses (very low number in MIMIC) and/or symptoms (totally absent in MIMIC), and the embeddings produced could be added as node features to enrich the graph (or as separated node types in a heterogeneous graph).

In conclusion, this two-stage multi-edge GNN framework represents a flexible and extensible approach for leveraging both intra-admission features and inter-admission relationships in EHR.

Further enhancements in edge modeling, multimodal data integration, and scalable graph architectures will continue to improve performance and broaden applicability across different healthcare settings.

Bibliography

[1] An, Y., Liu, Y., Chen, X., and Sheng, Y. (2022). *TERTIAN: Clinical Endpoint Prediction in ICU via Time-Aware Transformer-Based Hierarchical Attention Network*. Computational Intelligence and Neuroscience, 2022(1): 4207940.

[2] Arik S. Ö., Pfister T., "*TabNet: Attentive Interpretable Tabular Learning*", Advances in Neural Information Processing Systems, 2021.

[3] Bedoya, A. D., Clement, M. E., Phelan, M., Steorts, R. C., O'Brien, C., and Goldstein, B. A. (2019). Minimal impact of implemented early warning score and best practice alert for patient deterioration. *Critical Care Medicine*, 47(1): 49–55.

[4] Boll H. O., Amirahmadi A., Soliman A., Byttner S., Mendoza M. R., "*Graph Neural Networks for Heart Failure Prediction on an EHR-Based Patient Similarity Graph*", Institute of Informatics, Universidade Federal do Rio Grande do Sul, 2023.

[5] Bui H., Warrier H., Gupta Y., "*Benchmarking with MIMIC-IV, an Irregular, Sparse Clinical Time Series Dataset*", arXiv preprint arXiv:2401.00001, 2024.

[6] Cardoso, L. T., Grion, C. M., Matsuo, T., Anami, E. H., Kauss, I. A., Seko, L., and Bonametti, A. M. (2011). Impact of delayed admission to intensive care units on mortality of critically ill patients: a cohort study. *Critical Care*, 15(1): R28.

[7] Churpek, M. M., Wendlandt, B., Zadravecz, F. J., Adhikari, R., Winslow, C., and Edelson, D. P. (2016). Association between intensive care unit transfer delay and hospital mortality: a multicenter investigation. *Journal of Hospital Medicine*, 11(11): 757–762.

[8] Covington, P.; Adams, J.; and Sargin, E. 2016. Deep Neural Networks for YouTube Recommendations. In Proceedings of the 10th ACM Conference on

Recommender Systems, RecSys ’16, 191–198. New York, NY, USA: Association for Computing Machinery. ISBN 9781450340359.

[9] Daphne S., Rajam V. M. A., Hemanth P., Dinesh S., “*An Ensemble Patient Graph Framework for Predictive Modelling from Electronic Health Records and Medical Notes*”, Diagnostics, 2025.

[10] Darabi, S., Kachuee, M., Fazeli, S., and Sarrafzadeh, M. (2020). TAPER: Time-aware patient EHR representation. *IEEE Journal of Biomedical and Health Informatics*, 24(11): 3268–3275.

[11] Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers), 4171–4186.

[12] Fang, A. H. S., Lim, W. T., and Balakrishnan, T. (2020). Early warning score validation methodologies and performance metrics: a systematic review. *BMC Medical Informatics and Decision Making*, 20(1): 111.

[13] Goldberger, A. L., Amaral, L. A., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., Mietus, J. E., Moody, G. B., Peng, C.-K., and Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. *Circulation*, 101(23): e215–e220.

[14] Gorishniy Y., Rubachev I., Khrulkov V., Vetrov D., “*FT-Transformer: A Self-Attention Architecture for Tabular Data*”, International Conference on Learning Representations, 2022.

[15] Gupta S., Sharma S., Sharma R., Chandra J., “*Healing with Hierarchy: Hierarchical Attention Empowered Graph Neural Networks for Predictive Analysis in Medical Data*”, Under review or conference proceedings, 2025.

[16] Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. Advances in neural information processing systems, 30.

[17] Harutyunyan H., Khachatrian H., Kale D. C., Galstyan A., Greiner R., “*Multi-task learning and benchmarking with clinical time series data*”, arXiv preprint arXiv:1703.07771, 2017.

[18] Johnson, A. E., Bulgarelli, L., Shen, L., Gayles, A., Shammout, A., Horng, S., Pollard, T. J., Hao, S., Moody, B., Gow, B., et al. (2023). MIMIC-IV, a freely accessible electronic health record dataset. "*MIMIC-IV (version 3.1)*", PhysioNet, 2024. https://doi.org/10.13026/kpb9-mt58

[19] Kiekkas, P., Tzenalis, A., Gklava, V., Stefanopoulos, N., Voyagis, G., and Aretha, D. (2022). Delayed Admission to the Intensive Care Unit and Mortality of Critically Ill Adults: Systematic Review and Meta-Analysis. *BioMed Research International*, 2022(1): 4083494.

[20] Kipf, T. 2016. Semi-Supervised Classification with GraphConvolutional Networks. arXiv preprint arXiv:1609.02907.

[21] Li J., Wu B., Sun X., Wang Y., "*Causal Hidden Markov Model for Time Series Disease Forecasting*", arXiv preprint arXiv:2103.16391, 2021. https://arxiv.org/abs/2103.16391

[22] Liu X., Cheng J., Song Y., Jiang X., "*Boosting Graph Structure Learning with Dummy Nodes*", Proceedings of the 39th International Conference on Machine Learning, 2022.

[23] Maroudis C., Karathanasopoulou K., Stylianides C. C., Dimitrakopoulos G., Panayides A. S., "*Fairness-Aware Graph Neural Networks for ICU Length of Stay Prediction in IoT-Enabled Environments*", Under review or conference proceedings, 2024.

[24] Nagarajah, S., Krzyzanowska, M. K., and Murphy, T. (2022). Early warning scores and their application in the inpatient oncology settings. *JCO Oncology Practice*, 18(6): 465–473.

[25] Rajkomar A. et al., "*Scalable and accurate deep learning with electronic health records*", NPJ Digital Medicine, Vol. 1, p.18, 2018. https://www.nature.com/articles/s41746-018-0029-1

[26] Ren, W., Liu, Z., Wu, Y., Zhang, Z., Hong, S., Liu, H., and the MINDER Group. (2024). Moving beyond medical statistics: A systematic review on missing data handling in electronic health records. *Health Data Science*, 4: 0176.

[27] Rocheteau E., Tong C., Veličković P., Lane N., Liò P., "*Predicting Patient Outcomes with Graph Representation Learning*", arXiv preprint arXiv:2106.08159, 2021.

[28] Schlichtkrull M., Kipf T. N., Bloem P., van den Berg R., Titov I., Welling M., “*Relational Graph Convolutional Networks*”, Proceedings of the 2018 World Wide Web Conference, 2018.

[29] Shickel B., Tighe P. J., Bihorac A., Rashidi P., “*Multi-Task Prediction of Clinical Outcomes in the Intensive Care Unit using Flexible Multimodal Transformers*”, arXiv preprint arXiv:2111.05431, 2021.

[30] Siebra, C. A., Kurpicz-Briki, M., and Wac, K. (2024). Transformers in health: a systematic review on architectures for longitudinal data analysis. *Artificial Intelligence Review*, 57(2): 32.

[31] Singh, J., Sato, M., and Ohkuma, T. (2021). On missingness features in machine learning models for critical care: observational study. *JMIR Medical Informatics*, 9(12): e25022.

[32] Tong, C., Rocheteau, E., Veličković, P., Lane, N., and Liò, P. (2021). Predicting patient outcomes with graph representation learning. In *International Workshop on Health Intelligence*, 281–293. Springer.

[33] van de Water R., Schmidt H., Elbers P., et al., “*Yet Another ICU Benchmark: A Flexible Multi-Center Framework for Clinical ML*”, Under review or conference proceedings, 2025.

[34] Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y.; et al. 2017. Graph attention networks. stat, 1050(20): 10–48550.

[35] Wang Y., Li W., “*Integrating Multimodal EHR Data for Mortality Prediction in ICU Sepsis Patients*”, Statistics in Medicine, 2025.

[36] Wong, D. C.-W., Bonnici, T., Gerry, S., Birks, J., and Watkinson, P. J. (2024). Effect of Digital Early Warning Scores on Hospital Vital Sign Observation Protocol Adherence: Stepped-Wedge Evaluation. *Journal of Medical Internet Research*, 26: e46691.

[37] Wu Z., Pan S., Long G., Jiang J., Chang X., Zhang C., “*Connecting the Dots: Multivariate Time Series Forecasting with Graph Neural Networks*”, arXiv preprint arXiv:2005.11650, 2020. https://arxiv.org/abs/2005.11650

[38] Xu K., Li C., Tian Y., Sonobe T., Kawarabayashi K.-I., Jegelka S., “*Representation Learning on Graphs with Jumping Knowledge Networks*”, arXiv preprint arXiv:1806.03536, 2018.

[39] Ying Z., Bourgeois D., You J., Zitnik M., Leskovec J., “*GNNExplainer: Generating Explanations for Graph Neural Networks*”, Advances in Neural Information Processing Systems, 2019.