Bridging Graph and State-Space Modeling for Intensive Care Unit Length of Stay Prediction

Shuqi Zi[1] **Haitz Sáez de Ocáriz Borde**[2] **Emma Rocheteau**[1] **Pietro Liò**[1]

[1] University of Cambridge, UK
[2] University of Oxford, UK

Abstract

Predicting a patient's length of stay (LOS) in the intensive care unit (ICU) is a critical task for hospital resource management, yet remains challenging due to the heterogeneous and irregularly sampled nature of electronic health records (EHRs). In this work, we propose S^2G-Net, a novel neural architecture that unifies state-space sequence modeling with multi-view Graph Neural Networks (GNNs) for ICU LOS prediction. The temporal path employs Mamba state-space models (SSMs) to capture patient trajectories, while the graph path leverages an optimized GraphGPS backbone, designed to integrate heterogeneous patient similarity graphs derived from diagnostic, administrative, and semantic features. Experiments on the large-scale MIMIC-IV cohort dataset show that S^2G-Net consistently outperforms sequence models (BiLSTM, Mamba, Transformer), graph models (classic GNNs, GraphGPS), and hybrid approaches across all primary metrics. Extensive ablation studies and interpretability analyzes highlight the complementary contributions of each component of our architecture and underscore the importance of principled graph construction. These results demonstrate that S^2G-Net provides an effective and scalable solution for ICU LOS prediction with multi-modal clinical data. The code can be found at `https://github.com/ShuqiZi1/S2G-Net`.

1 Introduction

There has been a growing focus on predicting patient outcomes in intensive care units (ICUs), along with increasing research interest in applying Graph Deep Learning techniques for this purpose [53]. This surge in attention is largely driven by the expanded use of Electronic Health Records (EHRs), the demonstrated effectiveness of graph representation learning across diverse domains—including recommendation systems and misinformation detection [44], financial forecasting [66], and gene regulatory network modeling [11], to name a few—and the broader effort to reduce preventable deaths through early warning systems [45].

In this work, we bridge recent techniques in the Graph Neural Network (GNN) literature with sequence modeling methods by using graphs to model inter-patient correlations and sequence models to capture time-series patient data, enabling accurate predictions of ICU length of stay (LOS) for improved hospital resource allocation, management, and optimization. Despite recent progress, which we later discuss in Section 2.2, several challenges remain. Firstly, there is a clear gap in the literature for general-purpose models that jointly capture both temporal patient trajectories and inter-patient correlations, particularly for tasks like ICU LOS prediction, where existing approaches tend to focus on either temporal or spatial aspects in isolation or are tailored to narrow clinical outcomes such as mortality or disease progression. Secondly, traditional GNNs generally operate on single-view, static graphs, which are ill-suited for modeling heterogeneous, multi-modal clinical features and

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: The Second Workshop on GenAI for Health: Potential, Trust, and Policy Compliance.

dynamically evolving patient relationships. Transformer-based graph backbones (e.g., GraphGPS [50] and other Graph Transformers [58]) are powerful but computationally demanding, with quadratic complexity due to all-to-all node-wise communication, and they often fail to accommodate the unique characteristics of ICU data, such as sparsity and multi-view structure. Moreover, many existing models lack sufficient interpretability [43] (which is also critical in many real-world ML applications such as engineering [12] or finance [4]), limiting their clinical relevance.

In this study, we propose S^2G-Net (State-space × Optimized Multi-View GNN) for ICU LOS prediction. It comprises three components: a temporal pathway using a state-space backbone (Mamba [20]) to handle irregular and long-range temporal dependencies in patient time-series data; a graph pathway employing an optimized GraphGPS to learn from multiple patient similarity graphs built from diagnostic, semantic, and administrative views; and an auxiliary static-feature branch.

The main contributions of this work are:

- We propose a method for constructing heterogeneous patient similarity graphs using diagnostic, semantic, and administrative features, enabling richer representations of clinical relationships compared to single-view or static graphs.
- We develop a dual-path neural architecture that unifies temporal dynamics with population-level similarity modeling, tailored to the heterogeneous structure of ICU data.
- We adapt and enhance the GraphGPS architecture for clinical use, replacing Transformer [60] layers with state-space models (SSMs) and introducing more efficient graph fusion strategies.
- We benchmark S^2G-Net against strong sequence, graph, and hybrid baselines on the large-scale MIMIC-IV dataset [27], demonstrating superior performance in terms of accuracy, resource efficiency, and calibration.
- We conduct feature attribution, ablation, and calibration analyzes to evaluate the interpretability, reliability, and robustness of the model under real-world conditions.

2 Preliminaries and Related Work

2.1 Understanding the Complexities of Length of Stay Prediction

An ICU is a specialized healthcare unit providing advanced monitoring, physiologic organ support, and expert medical and nursing care for critically ill patients [41]. Globally, ICU resources are limited and expensive. Bed capacity varies by region, from 4.2 beds per 100,000 people in Portugal to 29.2 in Germany [52], with pandemics like COVID-19 pushing occupancy above 95% in some regions [18]. The daily cost per ICU patient cared for in developed countries ranges from $2,000 to $5,000 per day [6, 22, 5, 33], while prolonged hospital stays significantly increase the financial burden on both hospitals and families. In the United States itself, ICU costs contribute nearly 20 per cent of hospital costs [62], amounting to over $100 billion annually [30]. Beyond the costs, ICU LOS is closely tied to patient outcomes. Prolonged LOS is related to complications like hospital acquired infections [59], delirium, and muscle atrophy, while early discharge can lead to readmission or worsening of the prognosis [39]. Research has shown that in developed healthcare systems, ICU stays typically range from 3 to 8 days [42], but vary widely depending on the condition. For example, patients recovering from cardiac surgery may require a 1-2 day hospital stay, whereas patients with sepsis or acute respiratory distress syndrome (ARDS) may require weeks of intensive care. Accurate LOS prediction can help reduce inefficiencies, improve patient care, and alleviate the financial and operational burden on healthcare systems.

In fact, the dynamic and heterogeneous nature of ICU data makes predicting ICU LOS inherently challenging. ICU patients are continuously monitored, generating high-frequency time-series data including vital signs (e.g., heart rate, blood pressure) [3], laboratory results (e.g., blood gases, inflammatory markers), and interventions (e.g., medications, mechanical ventilation). These data display complex and intricate dependencies over time and across organ systems. For example, oxygenation may improve with diuretics in patients with heart failure, whereas worsening renal function may coincide with hypotension or sepsis progression, creating a highly non-linear and time-dependent problem. Moreover, in real ICU settings, only partial data points are recorded at uniform intervals, with physiological parameters like continuously monitored heart rate and oxygen saturation providing high-frequency data, while laboratory test results like blood gas levels or electrolyte

measurements are recorded intermittently. From a deep learning perspective, these characteristics present several challenges: ICU records are often irregular and incomplete, with certain variables sampled at widely varying frequencies. The data are also highly heterogeneous, varying across patients, conditions, and treatments. This makes it difficult for models to learn consistent patterns. Moreover, the relationships between variables are often non-linear and evolve over time, requiring models to capture both short- and long-term dependencies. These challenges are further complicated by the need for models to be accurate, interpretable, and robust in real-world clinical settings.

2.2 Deep Learning Approaches for Clinical Data

Temporal Sequence Models. Recurrent neural network (RNN) [54], particularly Long Short-Term Memory (LSTM) [26] models, are widely used for patient time-series modeling [37, 40, 49]. For example, Harutyunyan et al. [24] provided LSTM baselines for mortality, decompensation, length of stay, and phenotype prediction on MIMIC-III [28]. LSTMs can model sequential dependencies and have been adapted for multivariate and irregular EHR data using masking and imputation [36]. However, their performance declines with long and highly irregular ICU sequences. Recently, transformer architectures have shown promise in EHR modeling by capturing long-range dependencies through attention mechanisms. Models such as BEHRT [34] and Med-BERT [51] adapt transformers to structured clinical data, while recent approaches introduce positional encodings and sparse attention for handling irregular sampling [57]. Nonetheless, transformers are computationally intensive due to quadratic self-attention and can overfit with limited clinical data. Structured SSMs, such as S4 and Mamba, complement RNNs and transformers. S4 [21] achieves linear-time sequence modeling and maintains long-term dependencies using structured kernels. Mamba [20] introduces input-dependent recurrence for greater stability and efficiency on long sequences. Both have been applied to domains such as genomics and time-series analysis, but applications in high error rate ICU data remain rare.

Graph-Based Patient Modeling. Patient similarity graphs can model complex multi-modal relationships between individuals. Parisot et al. [46] applied graph convolutional networks (GCNs) to patient graphs for disease classification, constructing edges from demographic and imaging features. Subsequent studies have extended graph-based methods [32, 9] to tasks such as readmission prediction [17], medication recommendation [56], sepsis risk forecasting [64], as well as Alzheimer's disease and mild cognitive impairment prediction [10], demonstrating improvements over models that treat samples independently. Recent advances include dynamic and heterogeneous GNNs that capture temporal and attribute changes within patient cohorts. Dynamic GNNs model evolving edges and features over time [67], whereas multi-view GNNs integrate diverse data modalities (e.g., ICD codes, labs, text) for richer representations [31, 38]. Nevertheless, most clinical applications to date still use static or single-view graphs due to computational and modeling complexity. Beyond these considerations, another challenge lies in balancing local neighborhood aggregation and global context modeling. Local-Global GNNs such as GraphGPS [50] and GLGNN [14] address this issue by combining message passing with global attention modules. These approaches are well-suited to clinical graphs, where both local similarity and global cohort structure are relevant for prediction.

Hybrid Temporal-Graph Models. Models that jointly learn from temporal patient trajectories and inter-patient graph structure have shown improved performance in clinical forecasting tasks. GAMENet [56] integrates GCNs with dynamic memory modules for medication recommendation, and LSTM-GNN architectures [53] combine sequence modeling with graph-based neighborhood aggregation for ICU outcome prediction. Despite progress, most existing models inadequately integrate temporal and relational information, lack multi-modal graph construction, or are computationally inefficient. We address these gaps with S^2G-Net, a unified and efficient hybrid framework.

3 Methods

3.1 Problem Definition

We consider the task of predicting the ICU LOS for a set of patients using multi-modal clinical data collected during the first 48 hours of admission. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a patient similarity graph, where each node $v_i \in \mathcal{V}$ represents a unique ICU stay, and $(v_i, v_j) \in \mathcal{E}$ indicates clinical similarity based on diagnosis codes, semantic embeddings, or administrative metadata. Each node v_i is associated

with: (i) A multivariate time series $\mathbf{X}_i^{\text{TS}} \in \mathbb{R}^{T \times d}$, where $T = 48$ is the number of hourly time steps and d is the number of variables (labs, vitals); (ii) A static feature vector $\mathbf{x}_i^{\text{Flat}} \in \mathbb{R}^{d_f}$, formed by concatenating demographics, admission metadata, and diagnosis indicators; (iii) A continuous label $y_i \in \mathbb{R}_+$, representing the true ICU LOS in days. The objective is to learn a function F_Θ that maps patient data and graph structure to a LOS prediction: $\hat{y}_i = F_\Theta(\mathbf{X}_i^{\text{TS}}, \mathbf{x}_i^{\text{Flat}}, A(\mathcal{G}))$, where $A(\mathcal{G})$ is the adjacency matrix of the graph. Model parameters Θ are optimized to minimize the prediction error over all patients: $\min_\Theta \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(\hat{y}_i, y_i)$, where $\mathcal{L}(\cdot, \cdot)$ is a regression loss. All features are extracted strictly within the first 48 hours to ensure prospective prediction and avoid information leakage. LOS targets are transformed using $\log(1 + y_i)$ to facilitate stable learning, and predictions are reverted with $\exp(\hat{y}_i) - 1$.

3.2 Patient Graph Construction

To capture structured inter-patient relationships in ICU data, we construct a multi-view patient similarity graph $\mathcal{G}$ that integrates both structured diagnostic information and semantic similarity. Figure 1 illustrates the fusion process. The adjacency matrix for this graph serves as input to the model graph encoder and provides inductive relational bias by embedding both clinical and linguistic proximity among patients. The final graph defines four edge types: Type-0 for diagnosis-based similarity, Type-1 for BERT-based semantic similarity, and two optional augmentation types, Type-2 for minimum spanning tree (MST) and Type-3 for graph diffusion convolution (GDC).

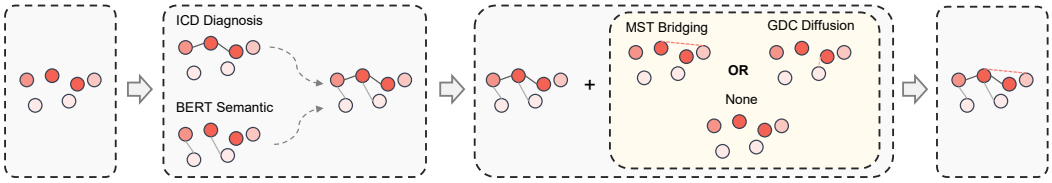

Figure 1: Multi-view graph construction and augmentation pipeline. Graphs based on ICD codes and BERT semantics are merged to form an initial patient graph. Connectivity and global structure are optionally enhanced via MST bridging or GDC. The final graph captures local, semantic, and long-range relations for downstream learning.

Diagnosis-Based Similarity Graph. We encode structured clinical similarity by constructing a graph from filtered ICD-9/10 codes. Each patient is represented by a sparse binary vector $\mathbf{d}_s \in \{0, 1\}^{d_d}$, indicating assigned specific diagnosis. Edges are constructed based on pairwise similarity using one of three strategies: (i) TF-IDF Cosine Similarity, where diagnosis vectors are reweighted by inverse document frequency and cosine similarity selects the top-k neighbors; (ii) FAISS Approximate KNN [29], which performs efficient inner product search over large patient sets; and (iii) Penalized Co-occurrence, which favors patients with high diagnostic overlap while penalizing multi-morbidity noise. After similarity calculation, we prune each node to its top-k_{diag} neighbors and normalize edge weights via $\log(1 + x)$ or z-score. We remove the bottom 30% of weak connections to improve sparsity. The resulting graph is denoted $\mathcal{G}_{\text{diag}} = (\mathcal{V}, \mathcal{E}_{\text{diag}})$.

BERT Semantic Graph. To capture semantic similarity beyond diagnosis code overlap, we construct a graph using sentence embeddings derived from patients' diagnosis descriptions. For each patient, diagnosis codes are concatenated into a single text sequence, truncated or zero-padded to 512 tokens. A pre-trained DistilBERT model [55, 13] encodes each input into a fixed-length embedding $\mathbf{b}_s \in \mathbb{R}^{768}$. Patient similarity is computed via Gaussian-kernelized Euclidean distance:

$$w_{ij}^{\text{bert}} = \exp\left(-\frac{||\mathbf{b}_i - \mathbf{b}_j||_2^2}{2\sigma^2}\right) \tag{1}$$

where σ is set to the median pairwise distance: $\sigma = \text{median}_{i \neq j}(|\mathbf{b}_i - \mathbf{b}_j|_2)/\sqrt{2}$, following [65]. Top-$k_{\text{bert}}$ neighbors are retained, and the resulting graph is $\mathcal{G}_{\text{bert}} = (\mathcal{V}, \mathcal{E}_{\text{bert}})$.

Multi-View Graph Fusion. The initial graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is formed by merging $\mathcal{E}_{\text{diag}}$ and $\mathcal{E}_{\text{bert}}$, retaining edge types for modality-aware message passing. To ensure global connectivity and capture

long-range dependencies [35], we optionally apply two rewiring strategies: MST-based bridging to link disconnected components, and GDC-based diffusion using first-order Personalized PageRank [15]. A maximum out-degree of 15 is enforced via stratified edge-type sampling to preserve diversity.

3.3 Model Architecture

As illustrated in Figure 2, S^{2}G-Net is a dual-path neural architecture consisting of temporal encoders and multi-view graph encoders, augmented with flat feature encoders for static attributes.

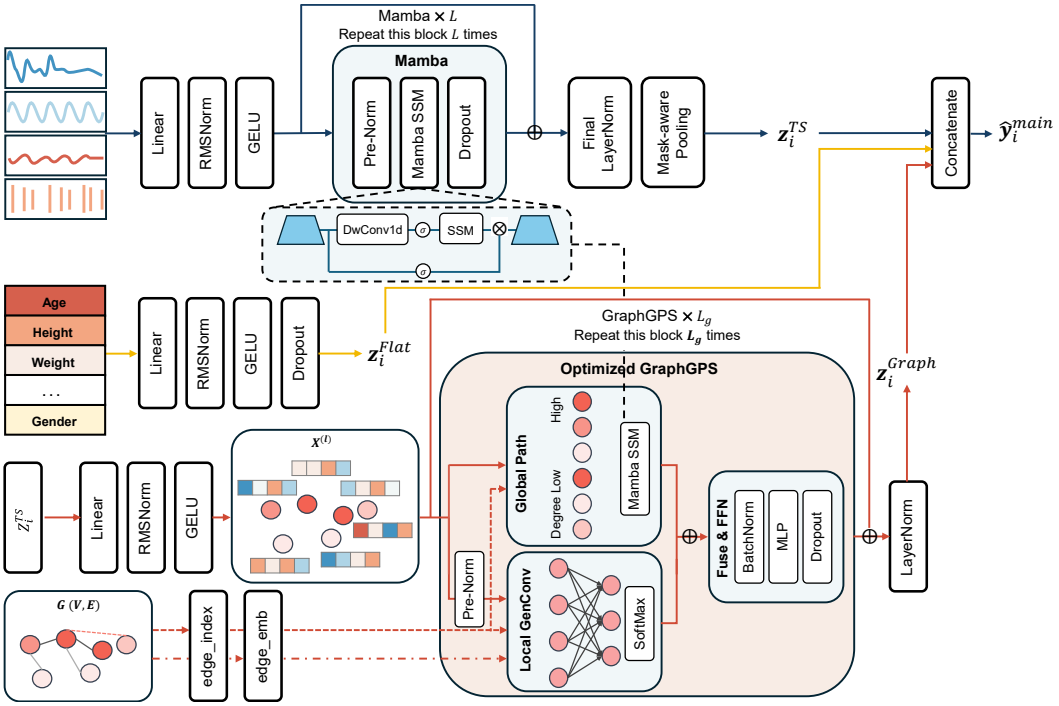

Figure 2: S^{2}G-Net model architecture. It consists of three branches: (i) a temporal encoder based on SSMs, (ii) a multi-view graph encoder capturing patient similarities, and (iii) a flat feature encoder for static attributes. The final representation is formed via weighted fusion of these components.

Temporal Encoder (State-Space). Time-series data in EHRs is typically long, sparse, and noisy. To capture long-range dependencies efficiently, we adopt the Mamba state-space model as the temporal encoder. Given a patient i with multivariate time-series $\mathbf{X}_i^{\text{TS}} \in \mathbb{R}^{T \times d}$ over the first 48 hours, we first apply a linear projection and normalization:

$$\mathbf{H}_i^{(0)} = \text{GELU}\left(\text{RMSNorm}\left(\mathbf{X}_i^{\text{TS}}\mathbf{W}_0 + \mathbf{b}_0\right)\right). \tag{2}$$

This sequence is then processed through L stacked Mamba blocks to model temporal patterns. To produce a fixed-length representation while handling missing values and variable lengths, we apply a mask-aware pooling function over the temporal dimension:

$$\mathbf{z}_i^{\text{TS}} = \text{MaskPool}\left(\mathbf{H}_i^{(L)}, \mathbf{m}_i\right) \tag{3}$$

where $\mathbf{m}_i \in \{0,1\}^T$ is a binary mask indicating observed time steps.

Graph Encoder (Local GENConv + Global Mamba). To capture population-level relational structure, we employ the multi-view patient similarity graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ introduced in Section 3.2. Nodes represent individual patients, and edges encode heterogeneous similarity relations with typed and weighted attributes. Each node i is initialized with its temporal embedding $\mathbf{z}_i^{\text{TS}}$, then projected via a two-layer MLP with LayerNorm to produce $\mathbf{x}_i^{(0)}$.

We stack L_g GraphGPS blocks. In each block $\ell \in \{0, \ldots, L_g - 1\}$, we compute a local update with GENConv layers that incorporate typed and weighted edge attributes, and a global update with a Mamba state-space encoder applied to a degree-ordered node sequence with slight random perturbations to resolve ties. We denote the local and global features by $\mathbf{h}_i^{(\ell)}$ and $\mathbf{g}_i^{(\ell)}$, respectively. The block update follows a residual pre-norm design:

$$\mathbf{u}_i^{(\ell)} = \mathrm{BN}\big(\mathbf{h}_i^{(\ell)} + \mathbf{g}_i^{(\ell)}\big), \tag{4}$$

$$\tilde{\mathbf{u}}_i^{(\ell)} = \mathrm{Dropout}\big(\mathrm{MLP}(\mathrm{LN}(\mathbf{u}_i^{(\ell)}))\big), \tag{5}$$

$$\mathbf{x}_i^{(\ell+1)} = \mathbf{x}_i^{(\ell)} + \tilde{\mathbf{u}}_i^{(\ell)}. \tag{6}$$

After L_g blocks we set the node representation to $\mathbf{z}_i^{\mathrm{Graph}} = \mathbf{x}_i^{(L_g)}$.

Static Feature Encoder. Static patient attributes $\mathbf{x}_i^{\mathrm{Flat}}$ are embedded by a dedicated feedforward module comprising linear projection, LayerNorm, GELU activation, and dropout. The resulting vector $\mathbf{z}_i^{\mathrm{Flat}}$ defined in Equation (7) is mapped into the shared latent space and provides complementary context to dynamic and relational features.

$$\mathbf{z}_i^{\mathrm{Flat}} = \mathrm{MLP}_{\mathrm{flat}}(\mathbf{x}_i^{\mathrm{Flat}}) \tag{7}$$

Fusion. To integrate information across temporal, relational, and static modalities, we perform a weighted concatenation of the corresponding embeddings. Let $\boldsymbol{\lambda} = (\lambda_{\mathrm{Graph}}, \lambda_{\mathrm{TS}}, \lambda_{\mathrm{Flat}})$ denote modality-specific importance weights, normalized via softmax. The fused representation is given by:

$$\mathbf{z}_i^{\mathrm{fused}} = \mathrm{Concat}\left(\lambda_{\mathrm{Graph}}\, \mathbf{z}_i^{\mathrm{Graph}},\ \lambda_{\mathrm{TS}}\, \mathbf{z}_i^{\mathrm{TS}},\ \lambda_{\mathrm{Flat}}\, \mathbf{z}_i^{\mathrm{Flat}}\right). \tag{8}$$

Prediction and Auxiliary Supervision. To reduce variance from outlier values and improve numerical stability, we *supervise in the log domain*. Define the monotone transform $T(y) = \log(1+y)$ and its inverse $T^{-1}(z) = \exp(z) - 1$. Let $\tilde{y}_i = T(y_i)$. The fused vector $\mathbf{z}_i^{\mathrm{fused}}$ is passed to a regression head that outputs a log-domain prediction $\tilde{\hat{y}}_i^{\mathrm{Main}}$; to encourage gradient flow, an auxiliary head on the time-series branch outputs $\tilde{\hat{y}}_i^{\mathrm{TS}}$. The training loss combines both outputs in the log domain:

$$\mathcal{L}_i = (1-\alpha)\, \mathcal{L}_{\mathrm{Huber}}\big(\tilde{\hat{y}}_i^{\mathrm{Main}},\, \tilde{y}_i\big) + \alpha\, \mathcal{L}_{\mathrm{Huber}}\big(\tilde{\hat{y}}_i^{\mathrm{TS}},\, \tilde{y}_i\big), \tag{9}$$

where $\alpha \in [0, 1]$ controls the auxiliary weight. To mitigate the influence of extreme LOS values, a sample-dependent reweighting scheme is applied (based on the original-domain label):

$$\mathcal{L}_i^{\mathrm{weighted}} = w(y_i)\, \mathcal{L}_i, \qquad w(y_i) = 1 + \gamma\, \mathbb{I}(y_i > \tau). \tag{10}$$

During evaluation, we convert predictions back to the original time domain via $\hat{y}_i = \max\{0,\, T^{-1}(\tilde{\hat{y}}_i^{\mathrm{Main}})\}$ and compute all metrics in that domain. Optimization is performed using AdamW with gradient clipping and early stopping based on validation R^2.

4 Experimental Validation

Data Description. We evaluate S^2G-Net on MIMIC-IV v3.1 dataset [27], a large-scale collection of de-identified ICU records from Beth Israel Deaconess Medical Center. It comprises 65,347 adult patients. We extract 216 features, including 174 time-series and 42 static, aligned to hourly bins within a 48-hour window. Missing values are imputed via forward fill and decay masking. The dataset is split patient-wise into train (70%), validation (15%), and test (15%).

Baselines. We compare our model with a baseline model suite covering three categories. (i) Sequential models include RNN [54], BiLSTM [19], and Transformer [60] for temporal sequence modeling, along with Mamba [20], a recent state-space model optimized for efficient long-range dependency learning. (ii) Graph-based models operate on static patient similarity graphs, including Graph Attention Network (GAT) [61], Message Passing Neural Network (MPNN) [16], GraphSAGE [23], and Optimized GraphGPS [50], which augments local message passing with global mixing via Mamba layers. (iii) Spatio-temporal models include LSTM-GNN [53], where patient time-series are encoded via LSTM and aggregated over static graphs, and dynamic LSTM-GNNs [53], which update graph structure during training to reflect evolving patient states. Results differ from Rocheteau et al. due to dataset (MIMIC-IV vs. eICU) and preprocessing differences but are used here as consistent baselines. For completeness, XGBoost [7] is included as a non-neural baseline for structured clinical prediction.

Evaluation Metrics. We report Mean Squared Error (MSE), Mean Squared Log Error (MSLE), Mean Absolute Deviation (MAD), log-transformed Mean Absolute Percentage Error (log-MAPE), R^2 score, and Cohen's linear weighted Kappa. MSE and R^2 are standard for regression; MSLE and log-MAPE are more robust to right-skewed LOS distributions. Kappa provides discrete agreement over stratified LOS bins. Formal definitions of all metrics are provided in Appendix C.

Setups. We tune hyperparameters with Optuna [1] over 75 trials, choosing the best configuration by validation R^2. The optimal setup adopts a 2-layer Mamba (128-dim) with a fusion coefficient of $\lambda = 0.5$. The patient similarity graph is built via FAISS using $k_{\text{diag}} = 3$, $k_{\text{BERT}} = 1$, and MST.

4.1 Results

S^{2}G-Net Performance. As shown in Table 1 and Figure 3, S^{2}G-Net consistently outperforms both neural and non-neural baselines across all 6 evaluation metrics. It obtains the highest R^2 (0.43 ± 0.01) with statistically significant improvements in R^2, MSE, and Kappa. Compared to the best-performing baseline GraphGPS, S^{2}G-Net improves R^2 by around 7.5% and reduces log-MAPE by 4.1%. It also delivers competitive results on MSLE and MAD, indicating that integrating spatio-temporal graph model with domain-specific priors may offer benefits for clinical prediction tasks.

Table 1: Performance comparison of S^{2}G-Net and baseline models. Mean$\pm$standard deviation over 5 runs. † indicates statistically significant improvement over the best baseline ($p < 0.05$). Red highlights best results; orange indicates second-best. $\uparrow$ indicates higher is better; $\downarrow$ lower is better.

Model	R^2 $\uparrow$	Kappa $\uparrow$	MSE $\downarrow$	MSLE $\downarrow$	MAD $\downarrow$	log-MAPE (%) $\downarrow$
Proposed Model						
S^{2}G-Net	$0.43 \pm 0.01^\dagger$	$0.42 \pm 0.00^\dagger$	$14.25 \pm 0.18^\dagger$	0.25 ± 0.01	1.88 ± 0.02	35.74 ± 1.24
Traditional Machine Learning Model						
XGBoost	0.32 ± 0.00	0.39 ± 0.00	17.32 ± 0.03	0.27 ± 0.00	1.96 ± 0.01	47.14 ± 0.17
Graph Neural Networks						
GraphGPS	0.40 ± 0.01	0.40 ± 0.00	15.08 ± 0.24	0.27 ± 0.02	1.93 ± 0.05	37.27 ± 1.26
GAT	0.29 ± 0.03	0.34 ± 0.02	17.81 ± 0.73	0.28 ± 0.01	2.02 ± 0.01	43.98 ± 3.76
MPNN	0.32 ± 0.00	0.36 ± 0.00	17.19 ± 0.04	0.28 ± 0.00	2.01 ± 0.00	41.20 ± 0.83
GraphSAGE	0.29 ± 0.03	0.34 ± 0.02	17.80 ± 0.64	0.28 ± 0.01	2.03 ± 0.01	43.70 ± 2.94
Sequence Models						
Mamba	0.33 ± 0.00	0.36 ± 0.00	16.89 ± 0.04	0.28 ± 0.02	2.03 ± 0.01	40.46 ± 1.03
Transformer	0.32 ± 0.01	0.36 ± 0.00	16.98 ± 0.10	0.29 ± 0.01	2.03 ± 0.01	44.80 ± 1.20
BiLSTM	0.31 ± 0.01	0.35 ± 0.01	17.23 ± 0.16	0.28 ± 0.00	2.01 ± 0.01	42.09 ± 2.38
RNN	0.26 ± 0.02	0.32 ± 0.01	18.53 ± 0.45	0.29 ± 0.01	2.02 ± 0.01	42.13 ± 2.69
Hybrid Temporal-Graph Models						
LSTM-MPNN	0.32 ± 0.03	0.35 ± 0.01	17.14 ± 0.68	0.29 ± 0.01	2.02 ± 0.02	42.73 ± 3.33
LSTM-GAT	0.31 ± 0.01	0.35 ± 0.01	17.36 ± 0.21	0.28 ± 0.00	2.01 ± 0.00	41.07 ± 1.58
LSTM-SAGE	0.32 ± 0.00	0.36 ± 0.00	17.14 ± 0.01	0.29 ± 0.00	2.02 ± 0.00	42.81 ± 1.98
DyLSTM-GAT	0.29 ± 0.01	0.33 ± 0.01	17.73 ± 0.27	0.28 ± 0.01	2.03 ± 0.00	42.83 ± 1.30
DyLSTM-MPNN	0.30 ± 0.01	0.34 ± 0.00	17.49 ± 0.29	0.28 ± 0.01	2.02 ± 0.00	44.77 ± 2.65
DyLSTM-GCN	0.31 ± 0.01	0.35 ± 0.00	17.41 ± 0.21	0.28 ± 0.01	2.03 ± 0.02	44.34 ± 3.74

Model Efficiency. Figure 3a, b shows that S^{2}G-Net achieves a balanced trade-off between performance and computational cost. Compared to models of similar scale, it achieves higher R^2 in comparable or shorter training times without significant parameter overhead (less than 2.5M), making it a computationally efficient alternative for deployment in resource-constrained clinical settings.

Ablation Study. Regarding modality contributions, excluding static features or using only static features each led to marked drops in model performance (Table 2). To obtain reliable predictions, both dynamic and static information sources are needed, and their integration is not trivially redundant.

Time window analysis reveals that longer observation windows provide substantial performance gains, especially between 6 and 24 hours (Figure 4a, d). Beyond 24 hours, the benefits plateau, possibly attributed to noise accumulation or reduced signal correlation in long-range dependencies.

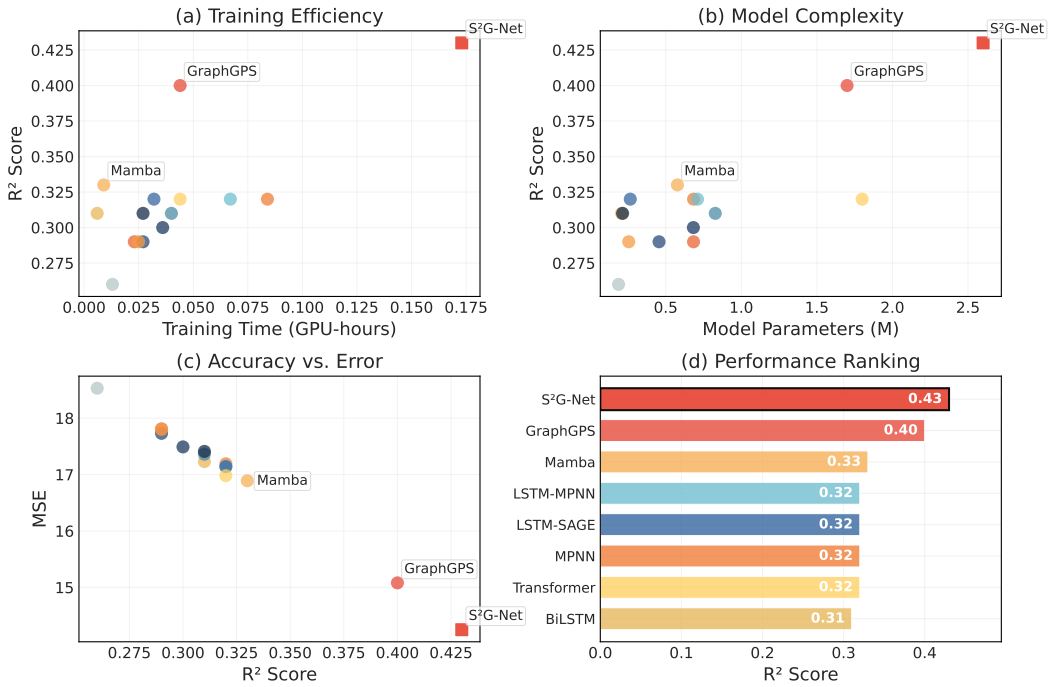

Figure 3: Multi-dimensional analysis of S^{2}G-Net and baselines. (a) Training cost vs. R^2; (b) Parameter count vs. R^2; (c) R^2 vs. MSE; (d) Overall ranking by R^2. XGBoost excluded for non-comparable computational metrics.

As illustrated in Figure 4b, e, Removal of physiological and vital sign features results in a notable performance drop. Among static features, ethnicity contributes the least to performance, while physiological and vital signals are more informative, as also verified by SHAP in Figure 5.

Table 2: Ablation study on temporal, feature, modality, and graph components. Mean $\pm$ standard deviation over 3 random seeds. $\uparrow$ indicates higher is better; $\downarrow$ lower is better. Missing values denote negligible or undefined agreement. Baseline refers to full model with 48h input and all features.

Model	**R^2** $\uparrow$	**Kappa** $\uparrow$	**MSE** $\downarrow$	**MSLE** $\downarrow$	**MAD** $\downarrow$	**log-MAPE (%)** $\downarrow$
Reference						
Baseline	0.43$\pm$0.00	0.42$\pm$0.00	14.25$\pm$0.00	0.25$\pm$0.00	1.88$\pm$0.00	35.86$\pm$0.12
Temporal Window Analysis						
Last 6h	0.28$\pm$0.00	0.33$\pm$0.00	18.02$\pm$0.03	0.39$\pm$0.00	2.15$\pm$0.00	45.09$\pm$0.21
Last 24h	0.40$\pm$0.00	0.40$\pm$0.00	15.09$\pm$0.01	0.28$\pm$0.00	1.95$\pm$0.00	37.77$\pm$0.13
Feature Group Impact						
No Physiology	0.39$\pm$0.00	0.38$\pm$0.00	15.40$\pm$0.00	0.31$\pm$0.00	2.12$\pm$0.00	42.03$\pm$0.32
No Vitals	0.40$\pm$0.00	0.39$\pm$0.00	15.16$\pm$0.00	0.28$\pm$0.00	2.00$\pm$0.00	40.24$\pm$0.27
No Ethnicity	0.42$\pm$0.00	0.41$\pm$0.00	14.21$\pm$0.01	0.26$\pm$0.00	1.91$\pm$0.00	36.17$\pm$0.12
Modality Analysis						
Static Only	–	–	34.69$\pm$1.87	1.32$\pm$0.44	3.12$\pm$0.35	93.66$\pm$10.73
No Static	–	–	33.43$\pm$0.00	0.99$\pm$0.00	2.91$\pm$0.00	105.54$\pm$17.01
Graph Robustness						
Drop 30%	0.43$\pm$0.00	0.41$\pm$0.00	14.35$\pm$0.00	0.25$\pm$0.00	1.90$\pm$0.00	36.90$\pm$0.11
Drop 50%	0.43$\pm$0.00	0.41$\pm$0.00	14.45$\pm$0.00	0.27$\pm$0.00	1.92$\pm$0.00	37.23$\pm$0.16
Drop 70%	0.38$\pm$0.00	0.37$\pm$0.00	15.63$\pm$0.00	0.29$\pm$0.00	2.03$\pm$0.00	42.67$\pm$0.31

Graph connectivity is further examined via controlled edge dropout (Figure 4c, f). Performance degrades gradually with increasing perturbation. The model retains a relatively stable R^2 up to 50%

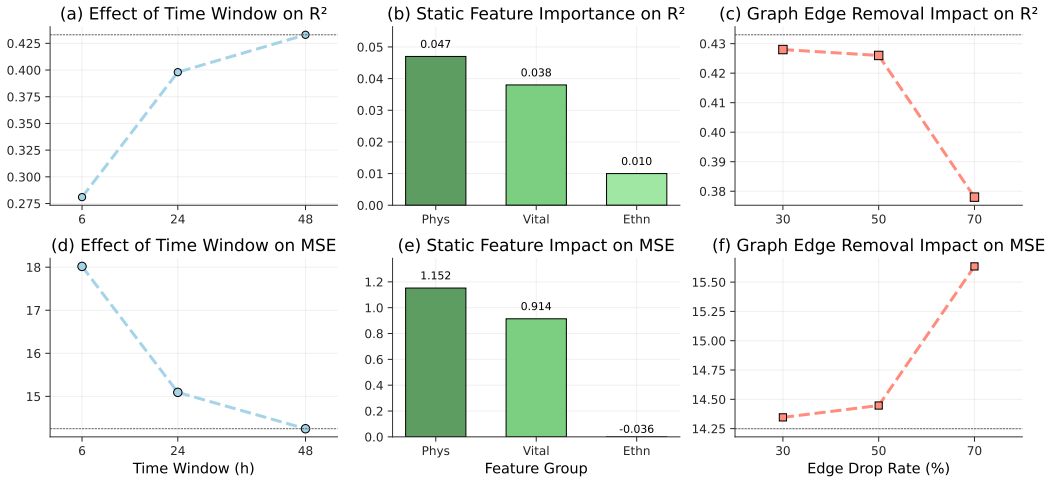

Figure 4: Ablation study on temporal, feature, and graph structure for R^2 and MSE. Dashed lines show the 48h baseline. (a,d) 6/24/48h input windows. (b,e) Removing static features Physiological (Phys), Vital signs (Vital), and Ethnicity (Ethn). (c,f) Edge dropout at 30%, 50%, and 70%.

edge removal. This indicate a degree of robustness to relational sparsity, a characteristic that is important when dealing with noisy or incomplete relationship data.

Interpretability. SHAP-based analysis revealed clinically meaningful predictors (Figure 5). The high contribution of GCS language and motor scores is consistent with previous studies, confirming that neurological status is the ICU prognosis strong predictor [2]. ICD codes related to acute respiratory failure (J96.2) and acute kidney injury (N17.0) also match common ICU admission diagnoses and mortality risk factors [47, 25]. The effect of ICU identifiers like MICU and SICU reflects differences in patient severity and treatment protocols by specialty. Its excellent calibration performance (ECE = 0.003) provides reliable predictive results for real-time clinical decision support.

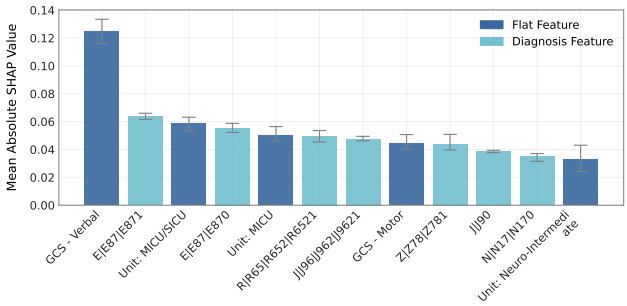

Figure 5: Top 12 features by SHAP importance, combining diagnosis-related and static features.

5 Conclusion

In this work, we introduced S^2G-Net, a novel dual-path neural architecture for ICU LOS prediction. Our model uniquely combines SSMs to capture temporal patient data and GNNs to represent inter-patient relationships, leveraging this as a geometric prior. The results demonstrated that this hybrid approach consistently outperformed existing baselines on the large-scale MIMIC-IV dataset, proving to be a highly effective and computationally efficient solution for a critical task in healthcare. Importantly, this research validates the importance of a holistic approach that considers both an individual patient's trajectory and relationships within the patient population. Future work will include external validation of S^2G-Net on independent cohorts to assess generalizability and extending it to broader clinical tasks and real-time decision support to enhance patient care and resource efficiency.

References

[1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2623–2631, 2019.

[2] Paulo G Bastos, Xiaolu Sun, Douglas P Wagner, Albert W Wu, and William A Knaus. Glasgow Coma Scale score in the evaluation of outcome in the intensive care unit: findings from the Acute Physiology and Chronic Health Evaluation III study. *Critical care medicine*, 21(10): 1459–1465, 1993.

[3] Cherisse Berry and David A. Spain. Monitoring and testing the critical care patient. MSD Manual Professional Edition, 2024. URL `https://www.msdmanuals.com/professional/critical-care-medicine/approach-to-the-critically-ill-patient/monitoring-and-testing-the-critical-care-patient`. Reviewed December 2022, modified April 2024.

[4] Damiano Brigo, Xiaoshan Huang, Andrea Pallavicini, and Haitz Saez de Ocariz Borde. Interpretability in deep learning for finance: a case study for the Heston model, 2021. URL `https://arxiv.org/abs/2104.09476`.

[5] Alayna Carrandi, Cheelim Liew, Matthew J. Maiden, Edward Litton, Colman Taylor, Kelly Thompson, and Alisa Higgins. Costs of Australian intensive care: A systematic review. *Critical Care and Resuscitation*, 2024.

[6] Binila Chacko, Nagarajan Ramakrishnan, and John Victor Peter. Approach to intensive care costing and provision of cost-effective care. *Indian Journal of Critical Care Medicine*, 27(12): 876, 2023.

[7] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[8] Jacob Cohen. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213, 1968.

[9] Haitz Sáez de Ocáriz Borde, Alvaro Arroyo, and Ingmar Posner. Projections of model spaces for latent graph inference. In *ICLR 2023 Workshop on Physics for Machine Learning*, 2023. URL `https://openreview.net/forum?id=Eu42jQQgDc`.

[10] Haitz Sáez de Ocáriz Borde, Anees Kazi, Federico Barbero, and Pietro Lio. Latent graph inference using product manifolds. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=JLR_B7n_Wqr`.

[11] Haitz Sáez de Ocáriz Borde, Anastasis Kratsios, Marc T. Law, Xiaowen Dong, and Michael M. Bronstein. Neural spacetimes for DAG representation learning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=skGSOcrIj7`.

[12] Haitz Sáez de Ocáriz Borde, David Sondak, and Pavlos Protopapas. Convolutional neural network models and interpretability for the anisotropic reynolds stress tensor in turbulent one-dimensional flows. *Journal of Turbulence*, 23(1-2):1–28, 2022. doi: 10.1080/14685248.2021.1999459. URL `https://doi.org/10.1080/14685248.2021.1999459`.

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pages 4171–4186, 2019.

[14] Moshe Eliasof and Eran Treister. Global-local graph neural networks for node-classification. *Pattern Recognition Letters*, 184:103–110, 2024.

[15] Johannes Gasteiger, Stefan Weißenberger, and Stephan Günnemann. Diffusion improves graph learning. *Advances in neural information processing systems*, 32, 2019.

[16] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. Pmlr, 2017.

[17] Sara Nouri Golmaei and Xiao Luo. DeepNote-GNN: Predicting hospital readmission using clinical notes and patient network. In *Proceedings of the 12th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, pages 1–9, 2021.

[18] Giacomo Grasselli, Antonio Pesenti, and Maurizio Cecconi. Critical care utilization for the COVID-19 outbreak in Lombardy, Italy: early experience and forecast during an emergency response. *Jama*, 323(16):1545–1546, 2020.

[19] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural networks*, 18(5-6):602–610, 2005.

[20] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. arXiv preprint arXiv:2312.00752, 2023. URL `https://arxiv.org/abs/2312.00752`.

[21] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021.

[22] Julian F. Guest, Tomas Keating, Dinah Gould, and Neil Wigglesworth. Modelling the annual NHS costs and outcomes attributable to healthcare-associated infections in England. *BMJ Open*, 10(1):e033367, 2020.

[23] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

[24] Hrant Harutyunyan, Hrant Khachatrian, David C. Kale, Greg Ver Steeg, and Aram Galstyan. Multitask learning and benchmarking with clinical time series data. *Scientific Data*, 6(1):96, 2019.

[25] Amarja Ashok Havaldar, EA Chinny Sushmitha, Sahad Bin Shrouf, Monisha HS, Madhammal N, and Sumithra Selvam. Epidemiological study of hospital acquired acute kidney injury in critically ill and its effect on the survival. *Scientific Reports*, 14(1):28129, 2024.

[26] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9 (8):1735–1780, 1997.

[27] Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, et al. MIMIC-IV, a freely accessible electronic health record dataset. *Scientific Data*, 10(1):1, 2023.

[28] Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. MIMIC-III, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.

[29] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547, 2019.

[30] Sneha Kannan, Jennifer Stevens, and Zirui Song. Growth in patient cost sharing for hospitalizations with and without intensive care among commercially insured patients. *Health Affairs*, 42 (9):1221–1229, 2023.

[31] Sein Kim, Namkyeong Lee, Junseok Lee, Dongmin Hyun, and Chanyoung Park. Heterogeneous graph learning for multi-modal medical data analysis. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 5141–5150, 2023.

[32] TN Kipf. Semi-supervised classification with Graph Convolutional Networks. *arXiv preprint arXiv:1609.02907*, 2016.

[33] Mehmet Kılıç, Nureddin Yüzkat, Celaleddin Soyalp, and Nurçin Gülhaş. Cost analysis on intensive care unit costs based on the length of stay. *Turkish Journal of Anaesthesiology and Reanimation*, 47(2):142, 2019.

[34] Yikuan Li, Shishir Rao, José Roberto Ayala Solares, Abdelaali Hassaine, Rema Ramakrishnan, Dexter Canoy, Yajie Zhu, Kazem Rahimi, and Gholamreza Salimi-Khorshidi. BEHRT: transformer for electronic health records. *Scientific Reports*, 10(1):7155, 2020.

[35] Huidong Liang, Haitz Sáez de Ocáriz Borde, Baskaran Sripathmanathan, Michael Bronstein, and Xiaowen Dong. Towards quantifying long-range interactions in graph machine learning: a large graph dataset and a measurement, 2025. URL `https://arxiv.org/abs/2503.09008`.

[36] Zachary C. Lipton, David C. Kale, Charles Elkan, and Randall Wetzel. Learning to diagnose with LSTM recurrent neural networks. *arXiv preprint arXiv:1511.03677*, 2015.

[37] Sicen Liu, Xiaolong Wang, Yang Xiang, Hui Xu, Hui Wang, and Buzhou Tang. Multi-channel fusion LSTM for medical event prediction using EHRs. *Journal of biomedical informatics*, 127: 104011, 2022.

[38] Zheng Liu, Xiaohan Li, Hao Peng, Lifang He, and S. Yu Philip. Heterogeneous similarity graph neural network on electronic health records. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 1196–1205. IEEE, 2020.

[39] Melina Loreto, Thiago Lisboa, and Viviane P. Moreira. Early prediction of ICU readmissions using classification algorithms. *Computers in Biology and Medicine*, 118:103636, 2020.

[40] Fenglong Ma, Radha Chitta, Jing Zhou, Quanzeng You, Tong Sun, and Jing Gao. Dipole: Diagnosis prediction in healthcare via attention-based bidirectional recurrent neural networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1903–1911, 2017.

[41] John C. Marshall, Laura Bosco, Neill K. Adhikari, Bronwen Connolly, Janet V. Diaz, Todd Dorman, Robert A. Fowler, Geert Meyfroidt, Satoshi Nakagawa, Paolo Pelosi, et al. What is an intensive care unit? A report of the task force of the World Federation of Societies of Intensive and Critical Care Medicine. *Journal of Critical Care*, 37:270–276, 2017.

[42] Vivek K. Moitra, Carmen Guerra, Walter T. Linde-Zwirble, and Hannah Wunsch. Relationship between ICU length of stay and long-term mortality for elderly ICU survivors. *Critical Care Medicine*, 44(4):655–662, 2016.

[43] Christoph Molnar, Giuseppe Casalicchio, and B. Bischl. Interpretable machine learning - a brief history, state-of-the-art and challenges. In *PKDD/ECML Workshops*, 2020. URL `https://api.semanticscholar.org/CorpusID:224706977`.

[44] Federico Monti, Fabrizio Frasca, Davide Eynard, Damon Mannion, and Michael M. Bronstein. Fake news detection on social media using geometric deep learning, 2019. URL `https://arxiv.org/abs/1902.06673`.

[45] Andrew Moon, J F Cosgrove, Dordi Lea, Abbie Fairs, and David M. Cressey. An eight year audit before and after the introduction of modified early warning score (MEWS) charts, of patients admitted to a tertiary referral intensive care unit after CPR. *Resuscitation*, 82 2:150–4, 2011. URL `https://api.semanticscholar.org/CorpusID:3399780`.

[46] Sarah Parisot, Sofia Ira Ktena, Enzo Ferrante, Matthew Lee, Ricardo Guerrero, Ben Glocker, and Daniel Rueckert. Disease prediction using graph convolutional networks: application to autism spectrum disorder and Alzheimer's disease. *Medical Image Analysis*, 48:117–130, 2018.

[47] Bryan D Park and Sarah Faubel. Acute kidney injury and acute respiratory distress syndrome. *Critical Care Clinics*, 37(4):835, 2021.

[48] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in Python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[49] Alvin Rajkomar, Eyal Oren, Kai Chen, Andrew M Dai, Nissan Hajaj, Michaela Hardt, Peter J Liu, Xiaobing Liu, Jake Marcus, Mimi Sun, et al. Scalable and accurate deep learning with electronic health records. *NPJ digital medicine*, 1(1):18, 2018.

[50] Ladislav Rampásek, Michael Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a general, powerful, scalable graph transformer. In *Advances in Neural Information Processing Systems*, volume 35, pages 14501–14515, 2022.

[51] Laila Rasmy, Yang Xiang, Ziqian Xie, Cui Tao, and Degui Zhi. Med-BERT: pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction. *NPJ Digital Medicine*, 4(1):86, 2021.

[52] Andrew Rhodes, Patrick Ferdinande, Hans Flaatten, Bertrand Guidet, Philipp G Metnitz, and Robert P Moreno. The variability of critical care bed numbers in Europe. *Intensive care medicine*, 38(10):1647–1653, 2012.

[53] Emma Rocheteau, Catherine Tong, Petar Veličković, Nicholas Lane, and Pietro Liò. Predicting patient outcomes with graph representation learning. *arXiv preprint arXiv:2101.03940*, 2021.

[54] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[55] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[56] Junyuan Shang, Cao Xiao, Tengfei Ma, Hongyan Li, and Jimeng Sun. Gamenet: Graph augmented memory networks for recommending medication combination. In *proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1126–1133, 2019.

[57] Huan Song, Deepta Rajan, Jayaraman Thiagarajan, and Andreas Spanias. Attend and diagnose: Clinical time series analysis using attention models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[58] Haitz Sáez de Ocáriz Borde. Elucidating graph neural networks, transformers, and graph transformers, 02 2024.

[59] Hui Jin Toh, Zhen Yu Lim, Philip Yap, and Terence Tang. Factors associated with prolonged length of stay in older patients. *Singapore Medical Journal*, 58(3):134, 2017.

[60] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[61] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, Yoshua Bengio, et al. Graph attention networks. *stat*, 1050(20):10–48550, 2017.

[62] Douglas P. Wagner, Thomas D. Wineland, and William A. Knaus. The hidden costs of treating severely ill patients: Charges and resource consumption in an intensive care unit. *Health Care Financing Review*, 5(1):81, 1983.

[63] Cort J Willmott and Kenji Matsuura. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate research*, 30 (1):79–82, 2005.

[64] Changchang Yin, Dakuo Wang, Jeffrey Caterino, and Ping Zhang. Sepsiscalc: Integrating clinical calculators into early sepsis prediction via dynamic temporal graph construction. *arXiv preprint arXiv:2501.00190*, 2025.

[65] Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. *Advances in neural information processing systems*, 17, 2004.

[66] Chao Zhang, Xingyue Pu, Mihai Cucuringu, and Xiaowen Dong. Graph neural networks for forecasting multivariate realized volatility with spillover effects, 2023. URL `https://arxiv.org/abs/2308.01419`.

[67] Yanping Zheng, Lu Yi, and Zhewei Wei. A survey of dynamic graph neural networks. *Frontiers of Computer Science*, 19(6):1–18, 2025.

A Compute Resources

All experiments are conducted using Python 3.8 with CUDA 11.8. For reproducibility, training and evaluation require access to GPUs such as NVIDIA RTX 3090/4090 or Tesla V100/A100, each with $\geq$12 GB of VRAM. A minimum of 32 GB of system memory is necessary to process the full MIMIC-IV dataset, and approximately 60 GB of local storage is used to accommodate raw and processed files. Experiments are executed under Linux environments, with Windows supported through WSL2.

B Dataset Statistics

Table 3: MIMIC-IV Cohort Statistics and Characteristics. Missing rate: 53.8% for height; Other Units include Neuro ICU, Neuro Stepdown, and miscellaneous units. SD: Standard deviation. Ranges contain extreme outliers from raw data (e.g., weight up to 5,864 kg, height up to 445 cm).

Dataset Overview			**ICU Unit Distribution**		
Characteristic	**Count**	**%**	**ICU Unit**	**Count**	**%**
Total ICU Stays	65,347	100.0	Medical ICU	12,732	19.5
Training Set	45,742	70.0	Cardiac Vascular ICU	11,520	17.6
Validation Set	9,802	15.0	Med./Surgical ICU	10,102	15.5
Test Set	9,803	15.0	Surgical ICU	8,980	13.7
Temporal Records	3,716,112	–	Trauma SICU	7,791	11.9
Static Features	42	–	Coronary Care Unit	7,127	10.9
			Other Units	7,095	10.9
ICU LOS Distribution			**Patient Demographics**		
Statistic	**Hours**	**Days**	**Characteristic**	**Value**	**Range/Dist.**
Mean ± SD	84.0 ± 122.4	3.5 ± 5.1	Age (years), mean ± SD	64.7 ± 17.1	18–103
Median	46.0	1.9	Male Gender	36,710	56.2%
25th Percentile	26.2	1.1	Female Gender	28,637	43.8%
75th Percentile	89.5	3.7	Weight (kg), mean ± SD	75.0 ± 39.4	1–5,864
Range	0.03–3,361.3	0.001–140.1	Height (cm), mean ± SD	169.3 ± 13.8	3–445

C Evaluation Metrics

Mean Squared Error (MSE). MSE quantifies the average squared difference between the predicted and true LOS values, providing a standard measure of prediction accuracy:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

where y_i and $\hat{y}_i$ denote the true and predicted LOS for patient i.

Mean Squared Logarithmic Error (MSLE) [48]. Given the right-skewed distribution of LOS in ICU cohorts, MSLE offers a more robust assessment by penalizing large relative errors while being less sensitive to extreme outliers:

$$\text{MSLE} = \frac{1}{N}\sum_{i=1}^{N}(\log(y_i + 1) - \log(\hat{y}_i + 1))^2$$

Mean Absolute Deviation (MAD) [63]. MAD measures the average absolute error, providing an interpretable, outlier-robust assessment:

$$\text{MAD} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$$

Log-transformed Mean Absolute Percentage Error (log-MAPE). Define $\ell_i = \log(1 + y_i)$ and $\hat{\ell}_i = \log(1 + \hat{y}_i)$. Then

$$\text{MAPE}_{\log 1p} = \frac{100\%}{N} \sum_{i=1}^{N} \frac{\left|\ell_i - \hat{\ell}_i\right|}{\max\{\ell_i, \epsilon\}},$$

where $\epsilon > 0$ avoids division by zero.

Coefficient of Determination (R^2) [48]. The R^2 score quantifies the proportion of variance in LOS explained by the model:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}$$

where $\bar{y}$ is the mean observed LOS.

Linear Weighted Kappa (Kappa) [8]. LOS predictions are discretized into 10 bins. The linear weighted kappa measures agreement between predicted and actual bins, accounting for chance agreement:

$$\kappa_w = \frac{p_o^w - p_e^w}{1 - p_e^w}$$

where p_o^w is the weighted observed agreement and p_e^w is the weighted expected agreement by chance. Linear weights are assigned based on bin distance:

$$w_{ij} = 1 - \frac{|i - j|}{k - 1}$$

where $k = 10$ bins and i, j are bin indices. This weighting gives full credit for exact agreement, partial credit for near predictions, and penalizes distant disagreements proportionally.

Table 4: Resource Consumption Comparison of Baseline Models. XGBoost is excluded as its computational metrics are not comparable to neural models.

Model	**Params**	**GPU-h**	**VRAM (GB)**
Proposed Method			
S^2G-Net	2.3M	0.173	6.4
Graph Neural Networks			
GraphGPS	1.7M	0.044	5.3
GAT	820.7K	0.023	2.6
MPNN	821.1K	0.084	2.0
GraphSAGE	306.2K	0.025	1.5
Sequence Models			
Mamba	692.9K	0.009	1.6
Transformer	1.8M	0.060	2.4
BiLSTM	249.3K	0.006	1.4
RNN	225.3K	0.013	1.3
Hybrid Temporal-Graph Models			
LSTM-MPNN	852.5K	0.067	2.0
LSTM-GAT	993.4K	0.040	3.2
LSTM-SAGE	318.6K	0.032	2.9
DyLSTM-GAT	546.9K	0.027	2.3
DyLSTM-MPNN	819.6K	0.036	2.5
DyLSTM-GCN	257.7K	0.027	1.8

D Efficiency and Resource Consumption

Table 4 analyzes the memory and computational trade-offs of each approach. S^2G-Net integrates both a temporal state-space encoder and a relational graph module, keeping practical resource consumption:

it requires 2.3M parameters, 0.173 GPU-hours per run, and a peak VRAM of 6.4 GB. This is modestly higher than GraphGPS (1.7M params, 5.3 GB VRAM) and substantially lower than ensemble or multi-stage architectures that are often infeasible in real-world deployments.

E Hyperparameter Sensitivity

We performed 75 independent trials using Optuna's tree-structured Parzen estimator (TPE) [1]. Figure 6a shows that validation R^2 increases steadily across 75 optimization trials, improving rapidly at first and tapering after approximately 12 trials. The performance histogram (Figure 6b) demonstrates a unimodal distribution, suggesting stable search behavior and the absence of degenerate hyperparameter regions.

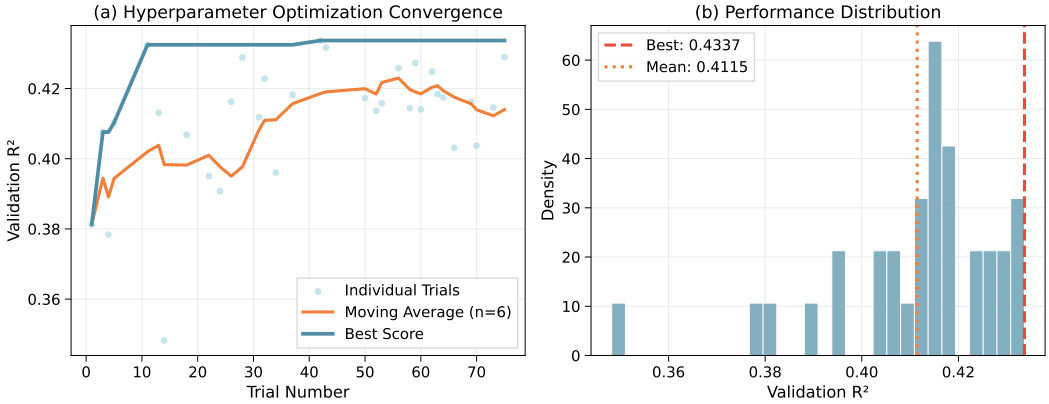

Figure 6: Hyperparameter optimization convergence. (a) Validation R^2 scores over 75 trials (excluding failures with $R^2 \leq 0$), including individual scores, an 8-trial moving average, and the best score. (b) Distribution of validation R^2 scores. The red dashed line marks the best score (0.4337), and the dotted line shows the mean (0.4115).

Figure 7 shows hyperparameter importance. The learning rate dominates (0.502), indicating the performance of S^2G-Net is highly sensitive to this setting, with suboptimal values causing marked degradation. Batch size (0.221) is the second most influential but far less critical. All other parameters, including fusion (λ), model depth, hidden/state dimensions, and regularization or pooling, each contribute under 6% to validation R^2 variance and can be tuned within broad ranges without substantial impact.

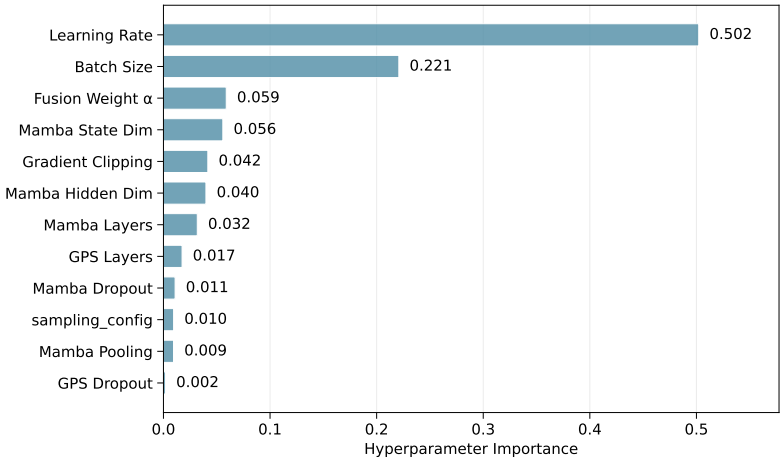

Figure 7: Hyperparameter sensitivity. Importance scores reflect each hyperparameter's impact on validation R^2 using Optuna's TPE.

F Hyperparameter Tuning

Table 5 shows that the optimal S^2G-Net model used a 2-layer Mamba encoder with 128 hidden dimensions and 16 state dimensions, a 2-layer GraphGPS module, dropout rates of 0.1, and last pooling in the time-series encoder. The fusion coefficient λ was set to 0.5, the learning rate to 1.7×10^{-4}, and the batch size to 32. The neighborhood sampling sizes are 15 and 10 for the first and second hops, respectively. The learning rate was sampled log-uniformly in $[1 \times 10^{-5}, 1 \times 10^{-3}]$ using Optuna's TPE; the best value selected by validation RMSE ($\mathrm{RMSE} = \sqrt{\mathrm{MSE}}$) was 1.7×10^{-4}. Hyperparameter optimization is conducted strictly on the validation set; the test set was used only for final model evaluation.

Table 5: Hyperparameter search space and the best-performing values selected by validation RMSE for S^2G-Net.

Parameter	**Search Space**	**Best Value**
Mamba d_model	64, 128, 256	128
Mamba layers	2, 3, 4	2
Mamba d_state	16, 32, 64	16
Mamba dropout	0.1, 0.2	0.1
Mamba pooling	mean, last	last
GPS layers	2, 3, 4	2
GPS dropout	0.1, 0.2	0.1
Fusion λ	0.3, 0.5, 0.7	0.5
Learning rate	log-uniform $[1 \times 10^{-5}, 1 \times 10^{-3}]$	1.7×10^{-4}
Batch size	32, 64, 128	32
Gradient clipping	0.0, 2.0, 5.0	2.0
Sampling config (N, L)	(15, 10), (25, 15)	(15, 10)

G Graph Construction Analysis

Table 6 shows 10 highest-scoring graph construction. The FAISS and TF-IDF methods are heavily weighted in the top results, while the MST reconnection strategy appears in high-ranking configurations. It indicates that S^2G-Net benefits from multi-view edges and judicious use of sparse connectivity, rather than indiscriminately increasing graph density. Moreover, small changes in k or similarity methods typically result in only minor differences in performance.

Table 6: Top 10 graph-construction settings ranked by R^2. Density is scaled by 10^{-6} for readability; $|E|$ denotes edge count.

k_{diag}	k_{BERT}	**Edge Method**	**Rewiring**	$\lvert E \rvert$	**Density**	$R^2 \uparrow$
3	1	FAISS	MST	206,847	96.88	0.4375
1	3	Penalize	None	237,179	111.09	0.4361
1	1	TF-IDF	MST	107,057	50.14	0.4321
1	3	FAISS	None	242,833	113.73	0.4304
1	1	FAISS	None	112,231	52.57	0.4293
5	1	FAISS	MST	294,081	137.74	0.4272
1	1	TF-IDF	None	107,057	50.14	0.4271
3	1	TF-IDF	None	186,078	87.15	0.4264
1	1	Penalize	None	106,614	49.93	0.4249
3	3	TF-IDF	GDC	316,726	148.34	0.4241

H Reliability and Calibration

We evaluated the probabilistic predictions of S^2G-Net using a calibration plot (Figure 8). The curve highly matches the actual LOS values within each interval, giving a standardized expected calibration

error of 0.003, showing that the calibration error is minimal. The accompanying histogram further confirms the good calibration of the prediction results within the clinically relevant LOS range.

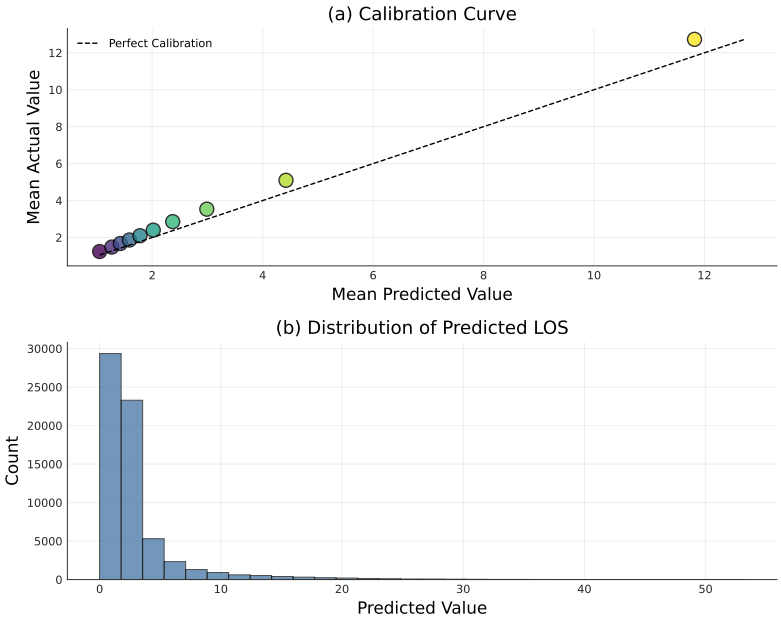

Figure 8: Reliability diagram comparing predicted and actual values across prediction bins, with a distribution histogram.