MEDUSA: GRAPH-CONDITIONED FLOW MATCHING FOR MISSING MODALITIES IN EHRS

Anonymous authors
Paper under double-blind review

ABSTRACT

Electronic health records (EHRs) are inherently multimodal, yet complete records are often unavailable because entire modalities may be missing. Existing approaches either reconstruct missing information or adapt prediction models to incomplete inputs, while graph based methods exploit relationships among patients but typically rely on deterministic message passing. We introduce MEDUSA, a graph conditioned generative framework that combines patient similarity with flow matching to impute missing modality representations. MEDUSA constructs a patient graph from observed modalities and learns a separate flow model for each maskable modality, allowing information from related patients to guide generation. Training follows an expectation maximization (EM) inspired alternating procedure that fits the generative models and jointly optimizes downstream prediction. Across multiple EHR prediction tasks, MEDUSA achieves strong predictive performance, particularly at higher missingness. Its imputations also better preserve the patient similarity structure of fully observed data than competing methods. These results show that combining relational structure with generative imputation is effective for learning from incomplete multimodal EHRs.

1 INTRODUCTION

Electronic health records (EHRs) are inherently multimodal, combining demographics, diagnostic codes, laboratory values, clinical notes, and other sources of clinical information (Shickel et al., 2018). While predictive models can benefit from these complementary modalities (Rajkomar et al., 2018), complete records are often unavailable because of acquisition costs, clinical workflows, or equipment and documentation failures (Le et al., 2025; Li et al., 2026). Robust prediction from EHRs therefore requires models that can operate effectively when entire modalities are missing (Zhang et al., 2022; Wu et al., 2024).

Existing approaches (Le et al., 2025) either recover missing modalities or their representations for prediction (Ma et al., 2021; Li et al., 2026), or adapt the prediction architecture to the subset of available inputs (Yao et al., 2024; Wu et al., 2024). More recently, graph-based methods have exploited patient–patient or patient–modality relationships to propagate information across incomplete records (Zhang et al., 2022; Wu et al., 2024). By connecting related patients or modalities, these approaches can recover information that is not available within a single incomplete record. However, they primarily rely on graph aggregation, fusion, or representation alignment, and therefore do not explicitly model a distribution over the missing modality representations.

Generative models provide a complementary approach by learning distributions over missing information rather than directly producing a deterministic reconstruction. They have been used for probabilistic imputation, including methods that incorporate patient similarity for temporal EHR completion (Tashiro et al., 2021; Dai et al., 2024), and for synthetic EHR generation under missing modalities (He et al., 2024). Yet these approaches do not directly address the setting considered here: generating entire missing modality representations using patient relational structure while jointly optimizing a downstream prediction task. This motivates a framework in which the patient graph conditions the generative process itself.

We introduce MEDUSA, a graph-conditioned generative framework that brings together patient graphs and generative modeling. MEDUSA constructs a patient similarity graph from jointly observed modalities and learns an independent flow-matching model (Lipman et al., 2023) for each

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

maskable modality. The graph conditions the generative process, allowing information from related patients and observed modalities to guide the imputation of missing latent representations. Training follows an expectation-maximization (EM)-inspired alternating optimization procedure: a generative pass fits the modality-specific flows, while a predictive pass regenerates missing embeddings and jointly optimizes the downstream task.

Our contributions are summarized as follows:

- We propose, to the best of our knowledge, the first formulation of missing modality completion in multimodal EHRs as the generation of modality representations conditioned on a patient similarity graph.
- We develop an EM-inspired alternating optimization procedure that couples modality-specific generation with downstream prediction.
- We evaluate MEDUSA on three EHR datasets against nine baselines. MEDUSA is particularly effective on MIMIC-IV and remains competitive across the evaluated tasks, with benefits becoming more pronounced under increasing modality missingness.

2 Related Work

Missing modalities in EHRs. Multimodal patient representation learning commonly assumes that all modalities are available for every patient, an assumption often violated in real-world EHRs (Zhang et al., 2022; Wu et al., 2024). Existing approaches broadly follow two strategies (Le et al., 2025). Imputation-based methods reconstruct missing modalities or their representations from observed ones, for instance, through cross-modal autoencoders (Ngiam et al., 2011) or modality priors (Ma et al., 2021). RAMCare combines retrieved case priors with patient-specific offsets for personalized modality recovery (Li et al., 2026). Missing modality-aware methods instead adapt the prediction architecture to the available inputs, for example, by selecting fusion strategies that improve robustness to missing modalities (Ma et al., 2022). However, these methods either produce a single point estimate of the missing representation or do not recover the missing information, only reducing its impact on prediction.

Graph-based methods provide another way to exploit incomplete multimodal observations by propagating information across patients and modalities. You et al. (2020) represent incomplete tabular data as a bipartite feature–sample graph. Chen & Zhang (2020) use a heterogeneous hypernode graph to model incomplete multimodal data. In healthcare, Zhang et al. (2022) propose M3Care, which combines modality-specific patient similarities into a shared graph adjacency and aggregates information from similar patients to impute missing modality representations in latent space. Wu et al. (2024) further introduce a patient–modality bipartite graph with a mutual-consistent contrastive objective for jointly handling missing modalities and labels. These approaches demonstrate the value of relational structure for incomplete multimodal data, but primarily use message passing, fusion, or representation alignment to handle missing information. MEDUSA instead uses patient relations to condition a generative model of missing modality representations.

Generative models for imputation. Generative models provide a complementary approach by learning distributions over plausible missing values rather than relying solely on deterministic reconstruction. Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020) have been successfully adapted to this setting. Tashiro et al. (2021) introduce CSDI, a conditional score-based diffusion model for probabilistic time-series imputation. Zheng & Charoenphakdee (2022) extend diffusion-based imputation to mixed-type tabular data, while Ouyang et al. (2023) address training diffusion models directly from incomplete tabular observations.

Patient similarity has also been incorporated into generative imputation. SADI (Dai et al., 2024) uses a similarity-aware diffusion model to capture dependencies across patients, features, and time for temporal EHR imputation. Its focus is missing values in temporal records, whereas MEDUSA generates entire missing modality representations conditioned on a patient similarity graph. FLEXGEN-EHR (He et al., 2024) addresses synthetic generation of heterogeneous EHRs with missing modalities, using optimal transport to align their feature spaces. MEDUSA instead targets completion of patient representations for downstream clinical prediction.

Most closely related to our optimization strategy, Zhang et al. (2025) propose DiffPuter, which alternates between training a diffusion model on current imputations and using the learned reverse process to refine missing values. DiffPuter operates on tabular feature vectors without explicitly modeling relations among samples. MEDUSA draws on this alternation but addresses a different setting: entire modalities are missing rather than individual features, and imputation serves downstream prediction rather than being the end goal. It generates missing modality representations using flow matching (Lipman et al., 2023) in a shared latent space, conditioned on a patient similarity graph constructed from the observed modalities.

3 METHODOLOGY

We introduce MEDUSA, a framework for imputing missing EHR modality embeddings through graph-conditioned flow matching. We first formulate the problem, then describe the patient graph, conditional imputation, and alternating generative and predictive optimization.

3.1 PROBLEM FORMULATION AND OVERVIEW

We consider an electronic health record (EHR) dataset with M modality groups. For patient i, $\mathbf{x}_i^{(m)}$ denotes the input associated with group m, and $a_i^{(m)} \in \{0, 1\}$ indicates whether that group is observed. We index demographics by $m = 1$ and assume they are always available; only non-demographic groups can be missing.

MEDUSA imputes missing modalities in a shared latent space rather than reconstructing raw inputs. Each modality is encoded as

$$\mathbf{h}_i^{(m)} = \mathbf{W}_m e_m\big(\mathbf{x}_i^{(m)}\big) \in \mathbb{R}^d, \tag{1}$$

where e_m is a modality-specific encoder and $\mathbf{W}_m$ maps its output to a common embedding dimension. A complete patient representation is

$$\mathbf{h}_i = \big[\mathbf{h}_i^{(1)} \| \cdots \| \mathbf{h}_i^{(M)}\big] \in \mathbb{R}^{Md}. \tag{2}$$

When group m is missing, MEDUSA estimates its latent representation $\hat{\mathbf{h}}_i^{(m)}$. Observed and imputed embeddings are concatenated into $\hat{\mathbf{h}}_i$ and used for downstream prediction.

MEDUSA imputes missing modality embeddings using graph-conditioned flow matching. From the observed modalities, we first construct a patient similarity graph. For each non-demographic group $m = 2, \ldots, M$, an independent flow $v_{\boldsymbol{\theta}_m}^{(m)}$ learns to transport Gaussian noise toward the distribution of that modality while conditioning on the patient graph. Cross-modal information therefore enters the flow through the graph, rather than through direct conditioning on the other modality embeddings. Training follows an EM-inspired alternating procedure. A generative pass fits the modality-specific flows using median-filled targets, while a prediction pass regenerates missing embeddings and optimizes the downstream task. Figure 1 summarizes the complete framework.

For a batch $\mathcal{B}$ of n patients, let $\mathbf{H}^{(m)} \in \mathbb{R}^{n \times d}$ stack the modality-m embeddings, $\mathbf{A} \in \mathbb{R}^{n \times n}$ denote the patient adjacency, and $\mathbf{a}^{(m)} \in \{0, 1\}^n$ collect the observation indicators. We use lowercase letters for scalars (except the modality count M), bold lowercase for vectors, and bold uppercase for matrices, retaining matrix notation for entries (e.g., $\mathbf{A}_{ij}$).

3.2 GRAPH-CONDITIONED MODALITY IMPUTATION

Patient graph. We construct the patient graph exclusively from observed modalities, as illustrated in block (2) of Figure 1. For each group m, we measure similarity between patients i and j using rescaled cosine similarity, defining the entries of $\mathbf{S}^{(m)}$ as

$$\mathbf{S}_{ij}^{(m)} = \frac{1}{2}\left(\frac{\langle \mathbf{h}_i^{(m)}, \mathbf{h}_j^{(m)} \rangle}{\|\mathbf{h}_i^{(m)}\|_2 \|\mathbf{h}_j^{(m)}\|_2} + 1\right). \tag{3}$$

The combined similarity matrix $\mathbf{S}$ aggregates only over modalities observed for both patients:

$$\mathbf{S}_{ij} = \frac{\sum_{m=1}^{M} a_i^{(m)} a_j^{(m)} \mathbf{S}_{ij}^{(m)}}{\sum_{m=1}^{M} a_i^{(m)} a_j^{(m)}}. \tag{4}$$

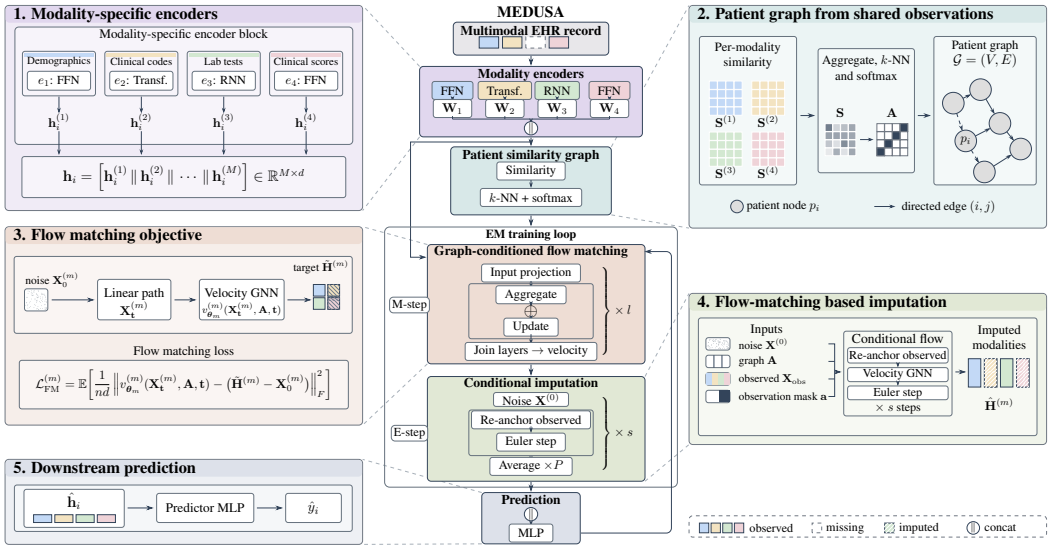


Figure 1: Overview of MEDUSA. The central workflow traces patient records through encoding, graph construction, flow-based imputation, and prediction, alternating M and E-steps during training. Modality-specific encoders (1) map observed inputs to shared embeddings, whose similarities over jointly observed modalities define a patient graph (2). Graph-conditioned flow matching (3) trains one flow for each of the $M-1$ non-demographic modalities, using observed or imputed targets. Euler sampling (4) imputes missing embeddings while re-anchoring observed ones, and an MLP (5) predicts from the completed patient representation.

The denominator is always non-zero because demographics are observed for every patient. For each patient i, we retain its k most similar patients, denoted by $\mathcal{N}(i)$, and normalize the retained similarities with a temperature-scaled softmax:

$$\mathbf{A}_{ij} = \begin{cases} \dfrac{\exp(\mathbf{S}_{ij}/\tau)}{\sum_{j'\in\mathcal{N}(i)} \exp(\mathbf{S}_{ij'}/\tau)}, & j \in \mathcal{N}(i), \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

The resulting directed graph contains no self-loops and is constructed only from observed embeddings; so imputed embeddings never influence which patients are connected or how strongly.

Flow matching. For each non-demographic group m, MEDUSA learns a separate graph-conditioned flow, corresponding to block (3) of Figure 1. Because some modality embeddings are missing during training, we construct complete surrogate targets during each generative pass. Observed rows retain their embeddings, while missing rows are filled with the coordinate-wise median of the observed embeddings of the same group:

$$\tilde{\mathbf{h}}_i^{(m)} = \begin{cases} \mathbf{h}_i^{(m)}, & a_i^{(m)} = 1, \\ \boldsymbol{\mu}_{\mathcal{B}}^{(m)}, & a_i^{(m)} = 0, \end{cases} \tag{6}$$

where $\boldsymbol{\mu}_{\mathcal{B}}^{(m)}$ is the batch median vector; stacking these rows gives $\tilde{\mathbf{H}}^{(m)}$. The median provides training targets for the flows. At imputation time, missing embeddings are generated by the flows, conditioned on similar patients in the graph.

For each patient, MEDUSA independently samples $\mathbf{x}_{0,i}^{(m)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ and $t_i \sim \mathcal{U}(0,1)$. We use the perturbed straight-line path

$$\mathbf{x}_{t_i,i}^{(m)} = (1-t_i)\mathbf{x}_{0,i}^{(m)} + t_i\tilde{\mathbf{h}}_i^{(m)} + \sigma_{\min}\boldsymbol{\xi}_i^{(m)}, \qquad \boldsymbol{\xi}_i^{(m)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d), \tag{7}$$

whose target velocity is $\tilde{\mathbf{h}}_i^{(m)} - \mathbf{x}_{0,i}^{(m)}$. Stacking these row vectors gives $\mathbf{X}_0^{(m)}$ and $\mathbf{X}_\mathbf{t}^{(m)}$, where $\mathbf{t} = (t_1, \ldots, t_n)^\top$. The modality-specific velocity field is trained using

$$\mathcal{L}_{\mathrm{FM}}^{(m)} = \mathbb{E}\left[\frac{1}{nd}\left\| v_{\boldsymbol{\theta}_m}^{(m)}\big(\mathbf{X}_\mathbf{t}^{(m)}, \mathbf{A}, \mathbf{t}\big) - \big(\tilde{\mathbf{H}}^{(m)} - \mathbf{X}_0^{(m)}\big)\right\|_F^2\right]. \tag{8}$$

Velocity network. Each velocity field is parameterized by a graph neural network (GNN) with trainable parameters $\boldsymbol{\theta}_m$. Let

$$\mathbf{E} = \mathrm{MLP}_{\mathrm{time}}^{(m)}(\mathbf{t}), \qquad \mathbf{Z}^{(0)} = \mathrm{MLP}_{\mathrm{in}}^{(m)}\big(\mathbf{X}_\mathbf{t}^{(m)}\big). \tag{9}$$

For l propagation layers, indexed by $\ell = 0, \ldots, l-1$, the network propagates information through

$$\mathbf{Z}_{\mathrm{agg}}^{(\ell)} = \mathbf{A}\mathbf{Z}^{(\ell)}, \tag{10}$$

$$\mathbf{Z}_{\mathrm{msg}}^{(\ell)} = \varphi_\ell^{(m)}\big([\mathbf{Z}^{(\ell)} \| \mathbf{Z}_{\mathrm{agg}}^{(\ell)} \| \mathbf{E}]\big), \tag{11}$$

$$\mathbf{Z}^{(\ell+1)} = \mathbf{Z}^{(\ell)} + \psi_\ell^{(m)}\big([\mathbf{Z}^{(\ell)} \| \mathbf{Z}_{\mathrm{msg}}^{(\ell)} \| \mathbf{E}]\big). \tag{12}$$

Here, $\varphi_\ell^{(m)}$ and $\psi_\ell^{(m)}$ are modality- and layer-specific affine transformations followed by a ReLU activation; $\mathbf{E}$ stacks the patient time embeddings. The velocity is then obtained from the representations at all propagation depths:

$$v_{\boldsymbol{\theta}_m}^{(m)}\big(\mathbf{X}_\mathbf{t}^{(m)}, \mathbf{A}, \mathbf{t}\big) = \mathrm{MLP}_{\mathrm{out}}^{(m)}\big([\mathbf{Z}^{(0)} \| \cdots \| \mathbf{Z}^{(l)} \| \mathbf{E}]\big). \tag{13}$$

Conditional sampling. The conditional sampling procedure is shown in block (4) of Figure 1. For a fixed modality m, we omit the modality index below. Let $\mathbf{X}_{\mathrm{obs}}$ contain the observed embeddings, with zeros at missing rows, and let $\mathbf{a}$ be its observation mask. Starting from a matrix $\mathbf{X}^{(0)}$ of independent standard Gaussian entries, we integrate the learned field using s Euler steps. For $t_r = r/s$ and $\Delta t = 1/s$,

$$\tilde{\mathbf{X}}^{(r)} = (\mathbf{1} - \mathbf{a}) \odot \mathbf{X}^{(r)} + \mathbf{a} \odot \Big[(1 - t_r)\mathbf{X}^{(0)} + t_r \mathbf{X}_{\mathrm{obs}}\Big], \tag{14}$$

$$\mathbf{X}^{(r+1)} = \tilde{\mathbf{X}}^{(r)} + \Delta t\, v_{\boldsymbol{\theta}_m}^{(m)}\big(\tilde{\mathbf{X}}^{(r)}, \mathbf{A}, t_r \mathbf{1}\big). \tag{15}$$

Here, $\mathbf{1}$ is the all-ones vector, and $\odot$ applies the mask row-wise. Before each Euler step, observed rows are reset to the path from their initial noise to their known embeddings; missing rows evolve according to the learned graph-conditioned velocity field. We repeat this procedure with independent initial noise and average the terminal states, denoting this average by $\overline{\mathbf{X}}$. The completed modality embeddings are

$$\hat{\mathbf{H}}^{(m)} = (\mathbf{1} - \mathbf{a}) \odot \overline{\mathbf{X}} + \mathbf{a} \odot \mathbf{X}_{\mathrm{obs}}. \tag{16}$$

Only missing rows are replaced by generated embeddings. Concatenating the completed modality matrices gives the batch representation $\hat{\mathbf{H}} \in \mathbb{R}^{n \times Md}$, with demographics retained unchanged.

3.3 Training and inference

MEDUSA uses an EM-inspired alternating optimization procedure with two passes per epoch: a generative pass (M-step) followed by an imputation/prediction pass (E-step). During the M-step, the modality-specific flows are optimized using

$$\mathcal{L}_{\mathrm{M}} = \sum_{m=2}^{M} \mathcal{L}_{\mathrm{FM}}^{(m)}. \tag{17}$$

The modality embeddings and patient graph are treated as fixed during this pass, such that only the flow parameters $\{\boldsymbol{\theta}_m\}_{m=2}^M$ are updated.

During the E-step, the flows regenerate the missing modality embeddings using the conditional sampling procedure of Section 3.2. The completed representation $\hat{\mathbf{H}}$ is passed to the predictor $f_{\boldsymbol{\phi}}$, and the downstream loss $\mathcal{L}_{\mathrm{E}} = \mathcal{L}_{\mathrm{pred}}(f_{\boldsymbol{\phi}}(\hat{\mathbf{H}}), \mathbf{y})$ updates the predictor, the modality encoders, and the flow models. Here, $\mathcal{L}_{\mathrm{pred}}$ denotes the prediction loss. Hence, the flows are optimized both

to match the modality distributions through $\mathcal{L}_{\mathrm{FM}}$ and to produce imputations for the downstream prediction task.

Gradients from $\mathcal{L}_{\mathrm{E}}$ propagate through the completed latent representations and the final velocity evaluation, while earlier Euler steps are detached. The differentiable edge weights of the graph provide an additional gradient path to the modality encoders, whereas k-NN selection remains non-differentiable. Thus, the predictive objective jointly trains the predictor, encoders, graph weights, and generative flows. Algorithm 1 in Appendix A summarizes the complete training procedure.

Inference. At inference, model parameters are fixed. Observed modalities are encoded and used to construct the patient similarity graph as during training. Missing modalities are then sampled from their graph-conditioned flows using the procedure in Section 3.2 and averaged to obtain imputed embeddings. These are combined with the observed embeddings into $\hat{\mathbf{H}}$, which is passed to $f_{\boldsymbol{\phi}}$ for the final prediction.

Computational complexity. Let C_g denote the cost of constructing the patient graph for a batch of n patients, and C_v the cost of one velocity network evaluation. Computing the pairwise similarities of Eq. (3) requires $\mathcal{O}(Mn^2d)$ operations, followed by neighbor selection Eq. (5), so $C_g = \mathcal{O}(Mn^2d)$. In the velocity network of Eq. (13), each of the l propagation layers multiplies $\mathbf{A}$ by an $n \times d_h$ matrix, where d_h is the hidden width, so $C_v = \mathcal{O}\big(l(n^2 d_h + n d_h^2)\big)$. The graph is constructed once per batch in each pass and reused across modalities and sampling steps. The M-step evaluates each of the $M-1$ flows once, and the E-step applies the Euler update of Eq. (15) s times along each of P trajectories, for each modality missing in the batch. Processing a batch in the two training passes together therefore costs $\mathcal{O}\big(C_g + (M-1)(1+Ps)C_v\big)$, excluding modality encoders and the prediction head. Only the final Euler step of each trajectory is differentiated, so the backward pass adds $\mathcal{O}\big((M-1)PC_v\big)$ and its memory cost does not depend on s. Sampling cost grows at most linearly with the number of maskable modalities, and linearly with the number of trajectories and Euler steps. For fixed batch size and model hyperparameters, per-epoch computation scales linearly with dataset size.

4 Experimental Evaluation

Datasets and tasks. We evaluate MEDUSA on two widely adopted EHR benchmarks, namely, MIMIC-IV (Johnson et al., 2024) and eICU (Pollard et al., 2019). We follow the preprocessing and splitting protocols of Wu et al. (2024) and retain only admissions with all modalities available, so that missingness can be simulated in a controlled way. Input modalities include demographics, clinical codes, and lab tests, complemented by discharge notes on MIMIC-IV and physiological variables on eICU. On both datasets, we address mortality and readmission prediction as binary classification tasks and report AUROC and AUPRC. Additional details on the datasets and tasks, performance metrics, and further experimental results, including results on EHRSHOT (Wornow et al., 2023), are reported in the Appendix (Sections B.1, B.4, and C, respectively).

Baselines and model selection. We compare MEDUSA against nine baselines (Section B.2) spanning simple imputation, graph-based feature imputation, generative imputation, and missing modality-aware multimodal learning: Mean, SMIL (Ma et al., 2021), FeatProp (Malitesta et al., 2024), GRAPE (You et al., 2020), M3Care (Zhang et al., 2022), MUSE and MUSE+ (Wu et al., 2024), RAMCare (Li et al., 2026), and DiffPuter (Zhang et al., 2025). All models share the same modality encoders (Section B.3) and a common set of fixed hyperparameters: 200 training epochs, batch size 512, dropout 0.25, no weight decay, and embedding dimension 128. We keep the ten checkpoints with the highest mean validation AUPRC over the ten masks and average their weights. The remaining hyperparameters are tuned via a trial search (Table 4), and we select the configuration with the highest validation AUPRC averaged over multiple missingness masks.

Missingness simulation. We simulate missing modalities under a Missing Completely At Random (MCAR) scheme. Each modality is dropped independently with probability $p = 0.1$. Demographics are always kept observed. For both the validation and test sets, we draw $K = 10$ random masks, which are shared across all methods so that every model faces identical missingness patterns. Methods that handle incomplete inputs are trained on masks resampled per batch with a fixed

Table 1: Test AUROC and AUPRC (%) on MIMIC-IV and eICU. Best results are shown in bold and second-best results are underlined.

Model	MIMIC-IV				eICU			
	Readmission		Mortality		Readmission		Mortality	
	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC
Mean	74.51±0.25	30.31±0.46	80.51±2.80	24.70±3.42	77.90±0.21	48.00±0.18	90.52±0.07	44.51±0.14
SMIL	75.09±0.21	31.49±0.40	80.67±2.68	24.94±2.76	78.01±0.08	48.00±0.06	90.70±0.06	45.03±0.14
FeatProp	75.13±0.07	31.10±0.24	82.86±0.07	27.13±0.48	78.10±0.03	47.92±0.07	90.44±0.40	45.12±0.68
GRAPE	75.65±0.32	31.47±0.74	83.91±0.39	28.91±0.66	78.13±0.35	47.85±0.56	89.92±0.05	43.47±0.19
M3Care	74.51±0.20	30.32±0.29	82.03±0.13	25.55±0.40	78.60±0.06	48.70±0.16	90.01±0.70	44.25±1.10
MUSE	75.28±0.02	31.40±0.33	83.22±0.61	28.58±1.07	78.34±0.26	48.56±0.27	90.98±0.20	45.98±0.77
MUSE+	75.04±0.58	31.09±0.56	82.88±0.35	27.88±0.07	78.32±0.03	48.59±0.20	90.54±0.58	44.86±0.84
RAMCare	75.78±0.07	31.84±0.18	83.07±0.30	26.91±0.95	78.00±0.95	48.60±1.01	**91.40**±0.05	**47.33**±0.34
DiffPuter	75.02±0.17	30.68±0.56	82.22±0.38	26.35±0.34	78.47±0.05	48.41±0.15	90.83±0.10	45.65±0.14
MEDUSA	**76.09**±0.11	**31.95**±0.28	**84.04**±0.10	**29.92**±0.45	**78.63**±0.21	**48.92**±0.11	90.72±0.29	45.74±1.23

seed. Inference-only imputation methods (Mean, FeatProp) are trained on fully observed data. Each model is trained with three random seeds and evaluated on the ten test masks. For each seed, we average the metrics over the ten masks, and we report the mean and standard deviation over the three seeds (see Section B.6).

Research questions. We structure our evaluation around three research questions:

RQ1) How does MEDUSA compare with existing methods in terms of predictive performance and reconstruction quality under missing modalities?

RQ2) What is the contribution of the main components of MEDUSA?

RQ3) What trade-off does MEDUSA achieve between predictive performance and computational efficiency?

4.1 Predictive performance and reconstruction quality (RQ1)

Table 1 compares predictive performance under missing modalities on MIMIC-IV and eICU.

Predictive performance. MEDUSA achieves the strongest results on MIMIC-IV across both tasks and evaluation metrics. The improvement is particularly clear for mortality, where MEDUSA provides the largest gain in AUPRC over the competing methods while also achieving the best AUROC. For readmission, the gains are more modest but remain consistent across both metrics. On eICU, MEDUSA remains competitive with the baselines, achieving the best readmission AUROC and near-best readmission AUPRC, while RAMCare performs better on mortality. Overall, MEDUSA shows its clearest gains on MIMIC-IV while remaining competitive across datasets and prediction tasks.

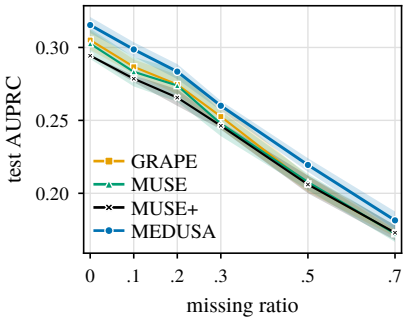
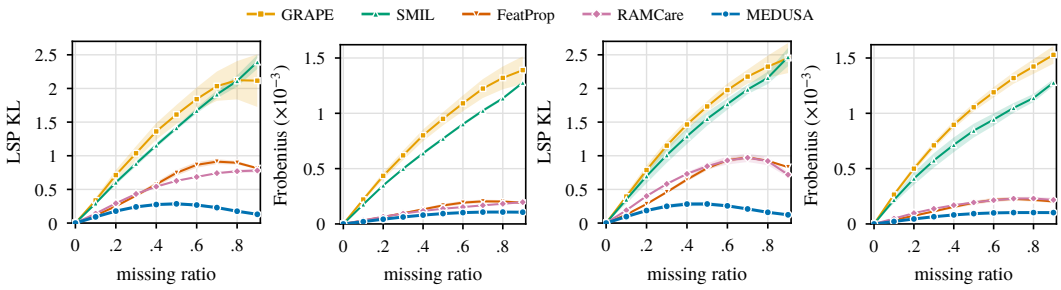


Figure 2: Test AUPRC as the missing modality ratio increases on MIMIC-IV mortality, against the three strongest baselines on this task (mean over three training seeds × two mask seeds, ±1 sd).

Robustness to increasing missingness. We next examine how predictive performance changes as the proportion of missing modalities increases. Figure 2 reports test AUPRC on MIMIC-IV mortality across missingness ratios ranging from 0 to 0.7. Performance decreases for all methods as less information becomes available, but MEDUSA consistently remains among the strongest methods across the evaluated range and maintains the highest AUPRC at the largest missingness ratios. This suggests that MEDUSA's predictive advantage persists as the amount of observed multimodal information decreases.



(a) Readm., LSP ↓ (b) Readm., Frobenius ↓ (c) Mortality, LSP ↓ (d) Mortality, Frobenius ↓

Figure 3: Effect of missingness ratio on the structural distance between the imputed and fully observed similarity graphs, on MIMIC-IV readmission (a, b) and mortality (c, d): local structure KL divergence (LSP) and similarity-preserving Frobenius distance. Lines are means over three training seeds × three mask seeds, bands ±1 sd. ↓ lower is better.

Reconstruction quality. Downstream performance alone does not reveal whether the generated modality embeddings recover the relational structure of the fully observed data. We therefore compare the patient similarity graph induced by the imputed representations with the graph obtained when all modalities are observed. Specifically, we measure local structural divergence using LSP KL and global similarity distortion using the Frobenius distance; both metrics are defined in Appendix B.5. Figure 3 reports these metrics for MIMIC-IV readmission and mortality as the missing modality ratio increases.

MEDUSA produces substantially smaller structural discrepancies than the competing methods across both tasks and both reconstruction metrics. Whereas the discrepancy of several baselines increases sharply with the missingness ratio, the graphs reconstructed from MEDUSA's imputed representations remain close to the fully observed reference graph. Together with the predictive results above, this indicates that MEDUSA not only generates embeddings that are useful for downstream prediction, but also better preserves the patient relationships present in the complete multimodal representation.

4.2 Ablation Studies (RQ2)

We evaluate the contribution of the main components of MEDUSA through four ablation variants:

- **w/o Graph** replaces the adjacency $\mathbf{A}$ of Eq. (5) with the identity matrix, removing information propagation across patients.
- **w/o EM** replaces the alternating generative and predictive optimization of Section 3.3 with a single-pass training procedure.
- **w/o Re-anchoring** performs conditional sampling without re-anchoring (Eq. (14)) observed modality embeddings along the flow trajectory.
- **Median** replaces flow-based imputation with coordinate-wise median imputation (Eq. (6)).

All variants are trained under the same experimental protocol, with hyperparameters re-tuned independently for each variant. To assess whether the contribution of each component changes with the severity of missingness, we report AUPRC on MIMIC-IV at modality missingness probabilities of 10%, 30%, and 50%. Results are shown in Table 2.

MEDUSA achieves the best or second-best result in most settings, with its advantage becoming more consistent as missingness increases. Replacing flow-based imputation with the median yields the largest and most consistent degradation, indicating that generative imputation is important for recovering informative modality representations. Removing the alternating optimization also reduces performance in most settings, particularly for mortality. The patient graph contributes more clearly at moderate and high missingness, while re-anchoring has a smaller overall effect and appears to act primarily as a stabilizing constraint during conditional sampling.

Table 2: Ablation study on MIMIC-IV under increasing missingness. Test AUPRC (%) with modality-missingness probabilities of 10%, 30%, and 50%. Best results are shown in bold and second-best results are underlined.

Model	Readmission			Mortality		
	10%	30%	50%	10%	30%	50%
MEDUSA w/o Graph	**32.49**±0.35	29.36±0.10	26.97±0.20	<u>30.31</u>±0.33	<u>26.73</u>±0.11	21.83±0.14
MEDUSA w/o EM	32.05±0.47	29.04±0.43	<u>27.09</u>±0.40	29.12±0.17	25.58±0.18	21.32±0.31
MEDUSA Median	31.91±0.28	28.70±0.15	26.64±0.23	28.32±0.63	24.66±0.02	20.80±0.44
MEDUSA w/o Re-anchoring	<u>32.41</u>±0.21	<u>29.38</u>±0.17	26.92±0.33	**30.48**±0.18	26.57±0.16	<u>21.86</u>±0.25
MEDUSA	31.95±0.28	**29.85**±0.59	**27.27**±0.66	29.92±0.45	**26.74**±0.79	**22.16**±0.65

4.3 Efficiency analysis (RQ3)

We finally assess the trade-off between predictive performance and computational cost. Figure 4 compares test AUPRC with average training time per epoch on MIMIC-IV mortality for MEDUSA and the strongest competing baselines. MEDUSA achieves the highest AUPRC among the compared methods while requiring less training time than several multimodal baselines. Although graph construction and flow-based sampling introduce additional computation relative to simpler imputation methods, the resulting training cost remains moderate relative to the observed predictive gains. Detailed training and inference times, including the corresponding AUROC and AUPRC results, are reported in Appendix C.2.

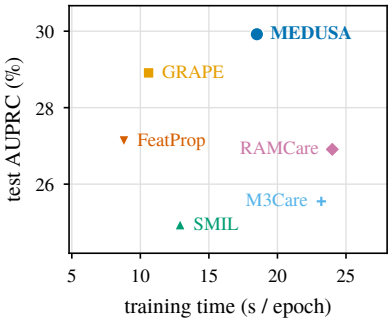


Figure 4: Test AUPRC versus training time (mean over three seeds).

5 Limitations

Despite its effectiveness in terms of prediction performance and efficiency, MEDUSA presents some limitations, both in its methodology and in our experimental setting. On the methodological side, MEDUSA imputes latent modality representations rather than the original clinical measurements. Our reconstruction analysis therefore assesses whether the relational structure among patients is preserved, not the clinical fidelity or interpretability of individual imputed features. A natural next step is to investigate whether, and to what extent, graph-conditioned generative models can operate on raw data and impute missing modalities directly in the original data space. On the experimental side, our evaluation mainly relies on controlled MCAR masking of originally complete patient records, which does not fully capture the structured or informative missingness observed in clinical practice. We report preliminary results under MNAR masking in Appendix C.3, and leave the evaluation under naturally occurring missingness to future work.

6 Conclusion

We introduced MEDUSA, a graph-conditioned generative framework for handling missing modalities in multimodal EHRs. MEDUSA combines patient-level relational information with modality-specific flow matching and an EM-inspired alternating optimization procedure, allowing missing latent representations to be generated while jointly optimizing the downstream prediction task. Across multiple EHR datasets and prediction tasks, MEDUSA achieves strong predictive performance, particularly on MIMIC-IV, and remains robust as the proportion of missing modalities increases. Our reconstruction analysis further shows that its generated representations better preserve the patient similarity structure of fully observed data than competing imputation methods. These results suggest that combining relational structure with generative modeling is a promising direction for learning from incomplete multimodal clinical data.

AI USE STATEMENT

Large language models were used solely to assist with the writing of this paper, namely to improve its clarity, fluency, and presentation. They did not contribute to the conception of the method, the development of the code, the design and execution of the experiments, or the interpretation of the results. The authors thoroughly checked and revised every part of the manuscript, and assume full responsibility for its content.

ETHICS STATEMENT

This work uses de-identified electronic health records from publicly available datasets (MIMIC-IV, eICU, EHRSHOT), accessed through the credentialed procedures required by their providers and used in accordance with the corresponding data use agreements. No new data were collected, no human participants were involved, and no attempt was made to re-identify any individual. MEDUSA is a research method for imputing missing values in multimodal clinical data and is not intended for direct clinical use: imputed values are statistical estimates, and relying on them for diagnostic or therapeutic decisions without thorough clinical validation could harm patients. Moreover, missingness in health records often reflects differences in access to care, so imputation models trained on such data may reproduce existing biases across patient subgroups; any downstream deployment should therefore assess performance separately for these subgroups and involve domain experts.

REPRODUCIBILITY STATEMENT

We describe all the elements needed to reproduce our results in the Appendix. The training procedure of MEDUSA is formalized in Algorithm 1 (Appendix A). Appendix B.1 details the datasets, the preprocessing and splitting protocols, and the definition of each prediction task, while Appendix B.3 describes the encoding of every modality, shared across all methods. The implementation of each baseline is reported in Appendix B.2, and the evaluation metrics in Appendices B.4 and B.5. Appendix B.6 specifies the simulation of missing modalities, including the fixed missingness masks shared by all methods, as well as the model selection and evaluation protocols, and Table 4 lists the fixed hyperparameters and the search spaces explored for every model. MIMIC-IV, eICU, and EHRSHOT cannot be redistributed, but they are available to researchers upon completion of the access procedures required by their providers. To ease replication, we release through an anonymous repository our codebase (Appendix D), including the preprocessing scripts, the generation of missingness masks, and the training and evaluation pipelines for MEDUSA and all baselines.

REFERENCES

Emily Alsentzer, John R. Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew B. A. McDermott. Publicly available clinical BERT embeddings. *CoRR*, abs/1904.03323, 2019. URL http://arxiv.org/abs/1904.03323.

Jiayi Chen and Aidong Zhang. HGMF: heterogeneous graph-based fusion for multimodal data with incompleteness. In *KDD*, pp. 1295–1305. ACM, 2020.

Zongyu Dai, Emily J. Getzen, and Qi Long. SADI: similarity-aware diffusion model-based imputation for incomplete temporal EHR data. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li (eds.), *International Conference on Artificial Intelligence and Statistics, 2-4 May 2024, Palau de Congressos, Valencia, Spain*, volume 238 of *Proceedings of Machine Learning Research*, pp. 4195–4203. PMLR, 2024. URL https://proceedings.mlr.press/v238/dai24c.html.

Huan He, William Hao, Yuanzhe Xi, Yong Chen, Bradley A. Malin, and Joyce C. Ho. A flexible generative model for heterogeneous tabular EHR with missing modality. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=W2tCmRrj7H.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html.

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Brian Gow, Benjamin Moody, Steven Horng, Leo Anthony Celi, and Roger Mark. MIMIC-IV. *PhysioNet*, October 2024. doi: 10.13026/kpb9-mt58. URL https://doi.org/10.13026/kpb9-mt58. Version 3.1.

Zhao Kang, Chong Peng, and Qiang Cheng. Kernel-driven similarity learning. *Neurocomputing*, 267:210–219, 2017. doi: 10.1016/J.NEUCOM.2017.06.005. URL https://doi.org/10.1016/j.neucom.2017.06.005.

Lien P. Le, Thu Nguyen, Michael A. Riegler, Pål Halvorsen, and Binh T. Nguyen. Multimodal missing data in healthcare: A comprehensive review and future directions. *Comput. Sci. Rev.*, 56: 100720, 2025. doi: 10.1016/J.COSREV.2024.100720. URL https://doi.org/10.1016/j.cosrev.2024.100720.

Bohao Li, Bowen Du, and Junchen Ye. Learning multimodal representations for incomplete ehrs with retrieval-augmented personalized modality recovery. *Inf. Fusion*, 133:104347, 2026. doi: 10.1016/J.INFFUS.2026.104347. URL https://doi.org/10.1016/j.inffus.2026.104347.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Mengmeng Ma, Jian Ren, Long Zhao, Sergey Tulyakov, Cathy Wu, and Xi Peng. SMIL: multimodal learning with severely missing modality. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 2302–2310. AAAI Press, 2021. doi: 10.1609/AAAI.V35I3.16330. URL https://doi.org/10.1609/aaai.v35i3.16330.

Mengmeng Ma, Jian Ren, Long Zhao, Davide Testuggine, and Xi Peng. Are multimodal transformers robust to missing modality? In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 18156–18165. IEEE, 2022. doi: 10.1109/CVPR52688.2022.01764. URL https://doi.org/10.1109/CVPR52688.2022.01764.

Daniele Malitesta, Emanuele Rossi, Claudio Pomo, Tommaso Di Noia, and Fragkiskos D. Malliaros. Do we really need to drop items with missing modalities in multimodal recommendation? In Edoardo Serra and Francesca Spezzano (eds.), *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, CIKM 2024, Boise, ID, USA, October 21-25, 2024*, pp. 3943–3948. ACM, 2024. doi: 10.1145/3627673.3679898. URL https://doi.org/10.1145/3627673.3679898.

Boris Muzellec, Julie Josse, Claire Boyer, and Marco Cuturi. Missing data imputation using optimal transport. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7130–7140. PMLR, 2020. URL http://proceedings.mlr.press/v119/muzellec20a.html.

Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. Multimodal deep learning. In *ICML*, pp. 689–696. Omnipress, 2011.

Yidong Ouyang, Liyan Xie, Chongxuan Li, and Guang Cheng. Missdiff: Training diffusion models on tabular data with missing values. *CoRR*, abs/2307.00467, 2023.

Tom Pollard, Alistair Johnson, Jesse Raffa, Leo Anthony Celi, Omar Badawi, and Roger Mark. eICU Collaborative Research Database. *PhysioNet*, April 2019. doi: 10.13026/C2WM1R. URL https://doi.org/10.13026/C2WM1R. Version 2.0.

Alvin Rajkomar, Eyal Oren, Kai Chen, Andrew M. Dai, Nissan Hajaj, Michaela Hardt, Peter J. Liu, Xiaobing Liu, Jake Marcus, Mimi Sun, Patrik Sundberg, Hector Yee, Kun Zhang, Yi Zhang, Gerardo Flores, Gavin E. Duggan, Jamie Irvine, Quoc Le, Kurt Litsch, Alexander Mossin, Justin Tansuwan, De Wang, James Wexler, Jimbo Wilson, Dana Ludwig, Samuel L. Volchenboum, Katherine Chou, Michael Pearson, Srinivasan Madabushi, Nigam H. Shah, Atul J. Butte, Michael D. Howell, Claire Cui, Gregory S. Corrado, and Jeffrey Dean. Scalable and accurate deep learning with electronic health records. *npj Digit. Medicine*, 1, 2018. doi: 10.1038/S41746-018-0029-1. URL https://doi.org/10.1038/s41746-018-0029-1.

Benjamin Shickel, Patrick James Tighe, Azra Bihorac, and Parisa Rashidi. Deep EHR: A survey of recent advances in deep learning techniques for electronic health record (EHR) analysis. *IEEE J. Biomed. Health Informatics*, 22(5):1589–1604, 2018. doi: 10.1109/JBHI.2017.2767063. URL https://doi.org/10.1109/JBHI.2017.2767063.

Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, pp. 2256–2265, 2015.

Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. CSDI: conditional score-based diffusion models for probabilistic time series imputation. In *NeurIPS*, pp. 24804–24816, 2021.

Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pp. 1365–1374. IEEE, 2019. doi: 10.1109/ICCV.2019.00145. URL https://doi.org/10.1109/ICCV.2019.00145.

Michael Wornow, Rahul Thapa, Ethan Steinberg, Jason A. Fries, and Nigam Shah. EHRSHOT: an EHR benchmark for few-shot evaluation of foundation models. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/d42db1f74df54cb992b3956eb7f15a6f-Abstract-Datasets_and_Benchmarks.html.

Zhenbang Wu, Anant Dadu, Nicholas J. Tustison, Brian B. Avants, Mike A. Nalls, Jimeng Sun, and Faraz Faghri. Multimodal patient representation learning with missing modalities and labels. In *ICLR*, 2024.

Yiding Yang, Jiayan Qiu, Mingli Song, Dacheng Tao, and Xinchao Wang. Distilling knowledge from graph convolutional networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pp. 7072–7081. Computer Vision Foundation / IEEE, 2020. doi: 10.1109/CVPR42600.2020.00710. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Yang_Distilling_Knowledge_From_Graph_Convolutional_Networks_CVPR_2020_paper.html.

Wenfang Yao, Kejing Yin, William K. Cheung, Jia Liu, and Jing Qin. Drfuse: Learning disentangled representation for clinical multi-modal fusion with missing modality and modal inconsistency. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan (eds.), *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2024, February 20-27, 2024, Vancouver, Canada*, pp. 16416–16424. AAAI Press, 2024. doi: 10.1609/AAAI.V38I15.29578. URL https://doi.org/10.1609/aaai.v38i15.29578.

Jiaxuan You, Xiaobai Ma, Daisy Yi Ding, Mykel J. Kochenderfer, and Jure Leskovec. Handling missing data with graph representation learning. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/dc36f18a9a0a776671d4879cae69b551-Abstract.html.

Chaohe Zhang, Xu Chu, Liantao Ma, Yinghao Zhu, Yasha Wang, Jiangtao Wang, and Junfeng Zhao. M3care: Learning with missing modalities in multimodal healthcare data. In Aidong Zhang and Huzefa Rangwala (eds.), *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, pp. 2418–2428. ACM, 2022. doi: 10.1145/3534678.3539388. URL https://doi.org/10.1145/3534678.3539388.

Hengrui Zhang, Liancheng Fang, Qitian Wu, and Philip S. Yu. Diffputer: Empowering diffusion models for missing data imputation. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=3fl1SENSYO.

He Zhao, Ke Sun, Amir Dezfouli, and Edwin V. Bonilla. Transformed distribution matching for missing value imputation. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 42159–42186. PMLR, 2023. URL https://proceedings.mlr.press/v202/zhao23h.html.

Shuhan Zheng and Nontawat Charoenphakdee. Diffusion models for missing value imputation in tabular data. In *NeurIPS 2022 First Table Representation Workshop*, 2022.

APPENDIX

TABLE OF CONTENTS

A Training Procedure **14**

B Experimental Settings **16**

B.1 Datasets and Tasks 16

B.2 Implemented baselines 16

B.3 Multimodal encoders 18

B.4 Performance metrics 19

B.5 Structural distance metrics 19

B.6 Experimental pipeline 19

B.7 Models' hyperparameter search 20

C Additional Experimental Results **20**

C.1 Results on EHRSHOT 20

C.2 Efficiency analysis 20

C.3 Missing-not-at-random evaluation 21

D Computational Resources and Codebase **22**

A TRAINING PROCEDURE

Algorithm 1 details one epoch of the alternating optimization described in Section 3.3. Sampling uses the modality-local notation of Section 3.2; observation masks are applied row-wise.

Algorithm 1 MEDUSA: one training epoch

Require: Minibatches $\mathcal{B}$ with observation indicators and labels; encoders $\{e_m, \mathbf{W}_m\}_{m=1}^{M}$; flows $\{v_{\boldsymbol{\theta}_m}^{(m)}\}_{m=2}^{M}$; predictor $f_{\boldsymbol{\phi}}$; neighbors k, temperature τ, Euler steps s, trajectories P, perturbation $\sigma_{\min}$

Generative pass (M-step)

for each minibatch $\mathcal{B}$ **do**
  Encode observed modalities as $\mathbf{H}^{(m)}$ ▷ Eq. (1)
  Construct $\mathbf{A}$ from observed embeddings ▷ Eqs. (3)–(5)
  for each modality $m = 2, \dots, M$ **do**
    Build median-filled targets $\tilde{\mathbf{H}}^{(m)}$ ▷ Eq. (6)
    Draw Gaussian noise $\mathbf{X}_0^{(m)}$ and independent patient times $t_i \sim \mathcal{U}(0,1)$
    Draw independent $\boldsymbol{\xi}_i^{(m)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ for each patient
    $\mathbf{x}_{t_i,i}^{(m)} \leftarrow (1-t_i)\mathbf{x}_{0,i}^{(m)} + t_i \tilde{\mathbf{h}}_i^{(m)} + \sigma_{\min}\boldsymbol{\xi}_i^{(m)}$ ▷ Eq. (7)
    Update only $\boldsymbol{\theta}_m$ using $\mathcal{L}_{\mathrm{FM}}^{(m)}$ ▷ Eq. (8)
  end for
end for

Predictive pass (E-step)

for each minibatch $\mathcal{B}$ **do**
  Encode observed modalities and rebuild $\mathbf{A}$
  Retain demographics: $\hat{\mathbf{H}}^{(1)} \leftarrow \mathbf{H}^{(1)}$
  for each modality $m = 2, \dots, M$ **do**
    Set $\mathbf{a} \leftarrow \mathbf{a}^{(m)}$ and form $\mathbf{X}_{\mathrm{obs}}$ with zeros at missing rows
    for $p = 1, \dots, P$ **do**
      Draw $\mathbf{X}_p^{(0)}$ with independent standard Gaussian entries
      for $r = 0, \dots, s-1$ **do**
        $t_r \leftarrow r/s$
        $\tilde{\mathbf{X}}_p^{(r)} \leftarrow (\mathbf{1}-\mathbf{a}) \odot \mathbf{X}_p^{(r)} + \mathbf{a} \odot [(1-t_r)\mathbf{X}_p^{(0)} + t_r \mathbf{X}_{\mathrm{obs}}]$ ▷ Eq. (14)
        $\mathbf{X}_p^{(r+1)} \leftarrow \tilde{\mathbf{X}}_p^{(r)} + \frac{1}{s} v_{\boldsymbol{\theta}_m}^{(m)}(\tilde{\mathbf{X}}_p^{(r)}, \mathbf{A}, t_r \mathbf{1})$ ▷ Eq. (15)
        if $r < s-1$ **then**
          Detach $\mathbf{X}_p^{(r+1)}$ from the computation graph
        end if
      end for
    end for
    $\overline{\mathbf{X}} \leftarrow \frac{1}{P}\sum_{p=1}^{P} \mathbf{X}_p^{(s)}$
    Complete $\hat{\mathbf{H}}^{(m)} \leftarrow (\mathbf{1}-\mathbf{a}) \odot \overline{\mathbf{X}} + \mathbf{a} \odot \mathbf{X}_{\mathrm{obs}}$ ▷ Eq. (16)
  end for
  Concatenate completed modalities: $\hat{\mathbf{H}} \leftarrow [\hat{\mathbf{H}}^{(1)} | \cdots | \hat{\mathbf{H}}^{(M)}]$
  $\mathcal{L}_{\mathrm{E}} \leftarrow \mathcal{L}_{\mathrm{pred}}\big(f_{\boldsymbol{\phi}}(\hat{\mathbf{H}}), \mathbf{y}\big)$ ▷ Section 3.3
  Update $\boldsymbol{\phi}$, the trainable encoders, and all $\boldsymbol{\theta}_m$ using $\mathcal{L}_{\mathrm{E}}$
end for

B EXPERIMENTAL SETTINGS

B.1 DATASETS AND TASKS

We evaluate our approach on three widely adopted EHR benchmarks for patient representation learning under missing modalities: MIMIC-IV (Johnson et al., 2024), eICU (Pollard et al., 2019), and EHRSHOT (Wornow et al., 2023). For all datasets, we follow the preprocessing and data splitting protocols of Wu et al. (2024). In addition, starting from the original set of patients and admissions, we retain only those admissions for which all modalities are available. This filtering step is essential, as it allows us to evaluate our models across a range of controlled, simulated missing modality scenarios rather than being restricted to the missingness patterns naturally occurring in the data (refer to Section B.6). Table 3 summarizes the statistics of each dataset after preprocessing, while Section B.3 provides further details on how each modality is encoded.

MIMIC-IV. This dataset comprises 40,535 patients admitted to the intensive care unit (ICU) of the Beth Israel Deaconess Medical Center in Boston, accounting for a total of 86,675 recorded admissions. In line with Wu et al. (2024), we consider four modalities: *demographics*, which groups age, gender, and ethnicity; *codes*, which gathers diagnoses, procedures, and medications; *lab tests*; and *discharge notes*, i.e., free-text clinical notes. Following Wu et al. (2024), we consider two prediction tasks: readmission and mortality. Specifically, the mortality task consists of predicting whether a patient dies within 90 days of discharge, while the readmission task consists of predicting whether the patient is readmitted within 15 days of discharge. Mortality labels are derived from the recorded date of death. When no date of death is available, we look at the patient's subsequent admission: if it occurs more than 90 days after discharge, the patient is labeled as alive; otherwise, the label is treated as missing. Readmission labels are obtained by comparing the discharge date with the date of the next admission. For each patient's final visit, the readmission label is set to missing, since the patient may have been transferred to another hospital.

eICU. This dataset contains clinical data collected from the ICUs of 207 hospitals across the United States, covering 112,002 patients and 137,487 admissions. Following Wu et al. (2024), we extract the same modalities used for MIMIC-IV, with the exception of discharge notes, which are not available. In addition, eICU provides physiological measurements and vital signs, which we include as a further modality (denoted as *apacheapsvar*). Similarly to MIMIC-IV, we follow again Wu et al. (2024) and consider the mortality and readmission tasks. The mortality task consists of predicting whether a patient dies before ICU discharge, while the readmission task consists of predicting whether the patient is readmitted to the ICU within 15 days during the same hospital stay. Both labels are derived from the current hospital stay, so no labels are missing.

EHRSHOT. This is a recently released EHR dataset comprising 3,060 patients from Stanford Medicine. Unlike MIMIC-IV and eICU, EHRSHOT offers a more longitudinal view of patient histories and is not limited to ICU stays. Consistently with the other two datasets, we consider *demographics* (age, gender, and ethnicity), *codes* (diagnoses, procedures, and medications), and *lab tests* as input modalities. The long length of stay task consists of predicting whether a patient's total length of stay during a visit will be at least 7 days. The labels are derived following the procedure and code by Wornow et al. (2023).

B.2 IMPLEMENTED BASELINES

We compare MEDUSA against nine baselines for missing information imputation in unimodal and multimodal learning, ranging from simple statistical heuristics to state-of-the-art approaches. Below, we outline the rationale of each method and the details of our implementation. Some baselines perform imputation only at inference time; unless otherwise stated, a method is applied at both training and inference time.

Mean. This inference-only baseline imputes each missing modality with the average of the available features for that modality. For categorical modalities, we use the median instead of the mean.

SMIL. Multimodal learning with severely missing modality (Ma et al., 2021) adopts a Bayesian meta-learning strategy that perturbs features in the latent space, so that each unimodal embedding approximates the corresponding complete multimodal representation. The learned perturbations are then used to reconstruct missing modalities from the available ones.

Table 3: Statistics of the cohorts: eICU and MIMIC-IV (readmission and mortality, one cohort per dataset) and EHRSHOT (long length of stay). Percentages are computed over all admissions (train, validation and test). LOS = length of stay, reported as median.

	eICU		**MIMIC-IV**		**EHRSHOT**
	Readmission	**Mortality**	**Readmission**	**Mortality**	**Long LOS**
Cohort					
Unit of prediction	ICU stay		Hospital admission		Inpatient visit
Hospitals	207		1		1
Patients	112,002		40,535		3,060
Admissions	137,487		86,675		5,200
Admissions per patient	1.23		2.14		1.70
Train / val / test	96,240 / 13,749 / 27,498		60,672 / 8,668 / 17,335		3,639 / 521 / 1,040
Positive (%)	15.73	4.61	12.97	6.37	24.48
LOS, days	1.8		3.1		4.0
Demographics					
Age, mean $\pm$ sd	62.3 ± 16.5		63.4 ± 15.7		56.9 ± 15.8
Age, median	64		65		59
$<$50 (%)	20.3		18.3		29.6
50–64 (%)	29.8		30.9		32.8
65–79 (%)	34.3		33.4		32.8
$\geq$80 (%)	15.6		17.5		4.9
Female (%)	45.4		50.6		48.8
Race (%)					
White	76.6		68.2		63.5
Black	11.2		18.2		3.9
Asian	1.6		2.8		13.3
Hispanic	3.9		5.6		15.0
Other	0.8		3.2		1.3
Unknown	5.9		2.0		18.0
Modalities					
Demographics (always observed)	age, gender, race		age, gender, race		age, gender, ethnicity
Codes (vocabulary size)	7,505		21,484		10,402
Lab tests (features)	158		114		7
Third modality	APACHE APS (36)		Discharge notes		–

FeatProp. Feature propagation (Malitesta et al., 2024) is an inference-only, graph-based imputation strategy that reconstructs missing features by propagating the available ones over a modality-specific similarity graph across multiple hops. Following the original formulation, we build each similarity graph by computing pairwise cosine similarities between samples within a modality, setting to zero the entries associated with samples missing that modality. Unlike the original method, however, we merge all modality-specific graphs into a single graph, which guarantees that every node has at least one neighbor (see Section B.6).

GRAPE. In its original formulation, GRAPE (You et al., 2020) models data as a bipartite graph connecting samples to their attributes, where a missing attribute corresponds to a missing edge. Imputation is thus cast as a link prediction task, in which a graph neural network (GNN) is trained to reconstruct the missing edges. Following Wu et al. (2024), we extend the method to the multimodal setting by building a bipartite graph between samples and modalities.

M3Care. Designed for missing modalities in multimodal healthcare data, M3Care (Zhang et al., 2022) imputes missing modalities by propagating the available ones over a multimodal similarity graph across multiple hops. The graph is constructed as in our implementation of FeatProp, but node similarities are computed through a learnable RBF kernel (Kang et al., 2017). Moreover, since propagation is performed by a trainable GNN, M3Care operates at both training and inference time. Finally, the model learns to attend over the resulting modality representations through a Transformer-like module.

MUSE. MUSE (Wu et al., 2024) is a mutual-consistent graph contrastive learning method that, similarly to GRAPE, represents data as a bipartite graph of samples and modalities, where missing edges denote missing modalities. In addition, MUSE introduces an unsupervised mutual-consistent contrastive loss, which aligns the embeddings of the same sample across different augmented views

of the bipartite graph, and a supervised counterpart, which pulls together the embeddings of samples sharing the same label.

MUSE+. This variant, proposed in the same work (Wu et al., 2024), extends MUSE with a mechanism that also handles samples with missing labels.

RAMCare. This model (Li et al., 2026) combines retrieval-augmented priors with a mixture-of-experts architecture to recover missing modalities in multimodal electronic health records (EHR) in two stages. First, it retrieves condition-matched cases from a co-clustered library of cross-modal pairs, which serve as priors for an initial reconstruction. Then, a mixture-of-experts module refines this reconstruction with a patient-specific offset.

DiffPuter. Zhang et al. (2025) employ diffusion models to impute missing values in unimodal tabular data. A key contribution is an iterative expectation-maximization procedure: in the M-step, a diffusion model is trained on the currently imputed dataset, while in the E-step the trained model is used to refine the imputed values. Since DiffPuter was originally designed for unimodal tabular data, we extend it to handle multimodal inputs for our experiments by concatenating modalities and treating them as tabular data. Notably, MEDUSA also draws inspiration from its iterative protocol, which we redesign for the multimodal graph setting of EHRs.

B.3 MULTIMODAL ENCODERS

To ensure a fair comparison between MEDUSA and all selected baselines, we adopt a shared encoding protocol for all modalities, partially inspired by the encoding strategy used in the MUSE pipeline (Wu et al., 2024). Regardless of its type, each modality is eventually projected into a latent space shared across modalities, and missing modalities are replaced with zero-valued vectors.

Categorical modalities. Categorical patient attributes, namely age, gender, and ethnicity, are grouped into a single *demographics* modality. We build a codebook that maps each category (e.g., *male*, *female*) to a unique ID, and represent each attribute through a dedicated lookup table of randomly initialized, learnable ID embeddings. During training, the embeddings of all demographic attributes are fused via element-wise summation into a single representation, which is then fed to a 2-layer MLP and an additional 1-layer MLP that projects it into the shared latent space. In our setting, demographics are assumed to be always observed (see Section B.6).

Mixed modalities. We refer to modalities combining categorical and continuous attributes as *mixed*; a representative example is the APACHE APS tabular data (`apacheapsvar`). Categorical attributes are one-hot encoded, and the resulting input is processed by a 2-layer MLP followed by a 1-layer MLP that projects it into the shared latent space.

Time-aware modalities. Time-aware attributes, namely event types and codes, are grouped into a single modality, referred to as *codes*, in which both are represented as categorical time series. Codes carry textual information: we encode them with a pre-trained language model (i.e., BioBERT (Alsentzer et al., 2019)[1]) and use the resulting representations to initialize frozen ID embeddings, which are subsequently projected into the shared latent space. Types, instead, are mapped to learnable ID embeddings, as done for the demographics modality. During training, type and code embeddings are fused via element-wise summation, and the resulting sequence is processed by a 2-layer Transformer encoder with 2 attention heads, which learns an attention-based fusion along the temporal dimension. A 1-layer MLP then projects the output into the shared latent space. Lab values (`labvectors`) constitute a second time-aware modality. They are encoded by a 1-layer recurrent neural network (RNN), which summarizes the entire sequence into a single representation, and are then projected into the shared latent space through a 1-layer MLP.

Pre-trained modalities. Discharge notes form a textual modality, which is first encoded with the same pre-trained language model (i.e., BioBERT (Alsentzer et al., 2019)). During training, the resulting embeddings are projected into the shared latent space through a 1-layer MLP.

[1] `emilyalsentzer/Bio_ClinicalBERT`.

B.4 PERFORMANCE METRICS

We frame the evaluation as a binary classification task. For each test instance i, the model outputs a score s_i for the positive class, which we compare against the ground-truth label $y_i \in \{0, 1\}$. Once predictions have been collected over the entire evaluation set, we derive the standard binary classification quantities from the counts of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN):

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \qquad \text{Recall} = \text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \qquad \text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}. \tag{18}$$

Building on these quantities, we report two threshold-independent metrics. The **AUROC** is the area under the ROC curve, which plots the TPR against the FPR. It can be interpreted as the probability that a randomly chosen positive instance receives a higher score than a randomly chosen negative one. The **AUPRC** is the area under the precision–recall curve. Unlike AUROC, it explicitly reflects class imbalance: a random classifier achieves a PR-AUC equal to the proportion of positive instances, rather than a fixed 0.5. This makes PR-AUC more informative when positives are rare.

B.5 STRUCTURAL DISTANCE METRICS

We measure how much a model's patient similarity graph changes when modalities go missing. Let $\mathbf{Z}$ be the patient representations that the model passes to its classifier after handling the missing modalities. Let $\mathbf{Z}^{\text{ref}}$ be the representations of the same patients when nothing is hidden. For each of them we build the similarity matrix $\mathbf{S}$ (resp. $\mathbf{S}^{\text{ref}}$) over n patients. We use two distances.

Local structure (LSP). Adapted from the local structure preserving distance of Yang et al. (2020), it compares the neighbourhood of each patient. Let $\mathcal{N}_k(i)$ be the k nearest neighbours of patient i in the reference graph. The distance is the mean over patients of

$$\text{KL}\Big[\text{softmax}\big(\mathbf{S}^{\text{ref}}_{i,\mathcal{N}_k(i)}/\tau\big) \,\Big\|\, \text{softmax}\big(\mathbf{S}_{i,\mathcal{N}_k(i)}/\tau\big)\Big],$$

i.e., how much the distribution of each patient over its neighbours changes.

Global structure (Frobenius). Following the similarity-preserving distance of Tung & Mori (2019), it compares the whole similarity matrix:

$$\frac{1}{n^2}\left\|\mathbf{G}^{\text{ref}} - \mathbf{G}\right\|_F^2,$$

where $\mathbf{G}$ is $\mathbf{S}$ with L2-normalized rows.

B.6 EXPERIMENTAL PIPELINE

Data splits and missingness simulation. We partition the patient observations into training, validation, and test sets. To simulate missing modalities, we adopt a Missing Completely At Random (MCAR) scheme, under which the absence of a modality is independent of both observed and unobserved data. Unless otherwise stated, each modality is independently dropped with probability $p = 0.1$; in the analyses of Section 4.1, we vary p across several missingness rates to assess the robustness of each method. By convention, we always keep at least one modality observed for every sample, typically the demographic one. For each sample in the validation and test sets, we draw $K = 10$ random missingness masks, each generated with a different random seed. This yields ten masked versions of the validation set, $\{\mathcal{M}^{\text{val}}_k\}_{k=1}^K$, and ten masked versions of the test set, $\{\mathcal{M}^{\text{test}}_k\}_{k=1}^K$. The same masks are shared across all methods, so that every model is evaluated under exactly the same missingness patterns.

Missingness at training time. Methods differ in whether they are exposed to missing modalities during training. Baselines that learn to handle incomplete inputs are trained on masked data, where a new random mask is sampled for each batch at runtime following the same MCAR scheme; the random generator is initialized with a fixed seed, so that the sequence of training masks is reproducible and identical across runs. Conversely, baselines that perform imputation only at inference time, such as Mean and FeatProp, are trained on the fully observed training data and encounter missing modalities only during evaluation.

Model selection. For each model, we run a hyperparameter search of 30 configurations, the first 15 sampled at random and the remaining 15 proposed by a tree structured Parzen estimator. On MIMIC-IV and eICU, the search is performed on the readmission task, and the selected configuration is reused for mortality. Each configuration θ is then evaluated on all ten masked validation sets, and its validation score is computed as the average AUPRC across masks:

$$\overline{\mathrm{AUPRC}}_{\mathrm{val}}(\theta) = \frac{1}{K}\sum_{k=1}^{K} \mathrm{AUPRC}\big(\theta; \mathcal{M}_k^{\mathrm{val}}\big). \tag{19}$$

The configuration achieving the highest average validation AUPRC, $\theta^* = \arg\max_\theta \overline{\mathrm{AUPRC}}_{\mathrm{val}}(\theta)$, is selected as the best one. Averaging over multiple masks makes the selection robust to the specific missingness pattern drawn, rather than favoring configurations that happen to perform well on a single realization.

Evaluation. Once θ^* is identified, we retrain the corresponding model on the training set with three different random seeds, and evaluate each trained model on each of the ten masked test sets. For each seed, every metric is averaged over the ten test masks. All results reported in the tables correspond to the mean and standard deviation of these per-seed averages across the three seeds.

Overall, this protocol is designed to make the training and evaluation of all baselines as similar and comparable as possible: every method is tuned with the same search procedure and seed, selected with the same criterion, and tested on the same set of missingness masks. As a result, differences in performance can be attributed to the methods themselves rather than to discrepancies in the experimental pipeline.

B.7 Models' hyperparameter search

To ensure a fair comparison, a subset of hyperparameters is kept fixed across all models. Every model is trained for 200 epochs with a batch size of 512; the final model averages the weights of the ten checkpoints with the highest validation AUPRC, averaged over the ten validation masks. We also fix the weight decay to 0 and the dropout rate to 0.25 for all methods, and we set the embedding dimension to 128 for all models that learn latent representations. The remaining hyperparameters are tuned through the search procedure described above, exploring the values reported in Table 4. Hyperparameters for which a single value is listed are kept fixed during the search.

C Additional Experimental Results

C.1 Results on EHRSHOT

Table 5 reports test AUROC and AUPRC on the long length of stay task of EHRSHOT, under the same protocol as in Section 4.1, for MEDUSA and the five strongest baselines on MIMIC-IV. MEDUSA achieves the best AUROC and AUPRC.

Table 5: Test AUROC and AUPRC (%) on EHRSHOT (length of stay). Best results are shown in bold and second-best results are underlined.

Model	AUROC	AUPRC
FeatProp	78.55±0.54	55.23±0.63
GRAPE	78.93±4.03	53.77±7.69
MUSE	79.66±0.16	55.03±0.28
MUSE+	<u>80.85</u>±0.16	<u>56.83</u>±0.67
RAMCare	80.42±0.52	55.50±1.14
MEDUSA	**81.28**±0.19	**56.90**±0.21

C.2 Efficiency analysis

Tables 6 and 7 complete the analysis of Section 4.3 for both tasks on MIMIC-IV and eICU. For each model, we report test AUROC and AUPRC together with the average training time per epoch and the average prediction time per sample. Predictive performance is reported as the mean and standard deviation over three training seeds, each averaged over the ten test masks. Times are measured in a dedicated run, with all models trained and evaluated on the same GPU model. Prediction time is computed over the whole test split, and therefore reflects batched throughput rather than the latency of a single patient.

Table 4: Hyperparameter values explored in the grid search for each baseline.

Model	Hyperparameter	Values
Mean	LR	0.0001, 0.0002, 0.001, 0.002
SMIL	LR	0.0001, 0.0002, 0.001, 0.002
	LAMBDA	0.001, 0.01, 0.1
	INNER_LR	0.001
	INNER_STEPS	1, 3, 5
	TEMPERATURE	4.0
FeatProp	LR	0.0001, 0.0002, 0.001, 0.002
	LAYERS	1, 2, 3, 4, 5
	TOP_K	10, 20, 30, 40
GRAPE	LR	0.0001, 0.0002, 0.001, 0.002
	ALPHA	0.01
	NUM_GNN_LAYERS	1, 2, 3
M3Care	LR	0.0001, 0.0005, 0.001, 0.005, 0.01
	LAMBDA	0.001
	ALPHA	0.01
	NUM_GNN_LAYERS	1, 2, 3
MUSE	LR	0.0001, 0.0002, 0.001, 0.002
	ALPHA	0.01
	NUM_GNN_LAYERS	1, 2, 3
MUSE+	LR	0.0001, 0.0002, 0.001, 0.002
	ALPHA	0.01
	NUM_GNN_LAYERS	1, 2, 3
RAMCare	LR	0.0001, 0.001, 0.002
	ALPHA	0.001, 0.01, 0.1
	RAM_CLUSTER_NUM	8, 16, 32
	RAM_HEAD_NUM	4, 8
DiffPuter	LR	0.0001, 0.0002, 0.001, 0.002
	DIFF_OPT_LR	0.0001, 0.0003, 0.001
	HIDDEN_DIM	1024
MEDUSA	LR	0.0001, 0.0002, 0.001, 0.002
	DIFF_OPT_LR	0.0001, 0.0003, 0.001
	NUM_GNN_LAYERS	1, 2, 3
	TOP_K	10, 20, 30, 40
	T	3, 5, 10

Table 6: Predictive performance and efficiency on MIMIC-IV. AUROC and AUPRC: mean ± sd over three seeds. Best results are shown in bold and second-best results are underlined.

Model	Readmission				Mortality			
	AUROC ↑ (%)	AUPRC ↑ (%)	Time ↓ per epoch (s)	Time ↓ per prediction (ms)	AUROC ↑ (%)	AUPRC ↑ (%)	Time ↓ per epoch (s)	Time ↓ per prediction (ms)
SMIL	75.09±0.21	31.49±0.40	13.2	<u>0.13</u>	80.67±2.68	24.94±2.76	12.9	<u>0.12</u>
FeatProp	75.13±0.07	31.10±0.24	**9.1**	0.18	82.86±0.07	27.13±0.48	**8.8**	0.17
GRAPE	75.65±0.32	31.47±0.74	<u>11.7</u>	0.14	<u>83.91</u>±0.39	<u>28.91</u>±0.66	<u>10.6</u>	<u>0.12</u>
M3Care	74.51±0.20	30.32±0.29	27.8	0.34	82.03±0.13	25.55±0.40	23.2	0.30
RAMCare	<u>75.78</u>±0.07	<u>31.84</u>±0.18	22.7	<u>0.13</u>	83.07±0.30	26.91±0.95	24.0	0.13
MEDUSA	**76.09**±0.11	**31.95**±0.28	18.6	**0.10**	**84.04**±0.10	**29.92**±0.45	18.5	**0.11**

C.3 MISSING-NOT-AT-RANDOM EVALUATION

The MCAR scheme of Section B.6 assumes that the absence of a modality is independent of the data. To assess robustness to informative missingness, we evaluate the trained models on test masks generated under a missing-not-at-random (MNAR) scheme. We adopt the logistic MNAR mechanism of Muzellec et al. (2020), as implemented by Zhao et al. (2023), applied at the level of modali-

Table 7: Predictive performance and efficiency on eICU. AUROC and AUPRC: mean ± sd over three seeds. Best results are shown in bold and second-best results are underlined.

Model	Readmission				Mortality			
	AUROC ↑ (%)	AUPRC ↑ (%)	Time ↓ per epoch (s)	Time ↓ per prediction (ms)	AUROC ↑ (%)	AUPRC ↑ (%)	Time ↓ per epoch (s)	Time ↓ per prediction (ms)
SMIL	78.01±0.08	48.00±0.06	22.3	<u>0.15</u>	90.70±0.06	45.03±0.14	23.0	<u>0.15</u>
FeatProp	78.10±0.03	47.92±0.07	**21.4**	0.20	90.44±0.40	45.12±0.68	**21.5**	0.20
GRAPE	78.13±0.35	47.85±0.56	<u>22.0</u>	**0.14**	89.92±0.05	43.47±0.19	<u>22.0</u>	**0.14**
M3Care	<u>78.60</u>±0.06	<u>48.70</u>±0.16	55.4	0.46	90.01±0.70	44.25±1.10	55.4	0.46
RAMCare	78.00±0.95	48.60±1.01	56.8	0.16	**91.40**±0.05	**47.33**±0.34	61.6	0.16
MEDUSA	**78.63**±0.21	**48.92**±0.11	27.7	0.20	<u>90.72</u>±0.29	<u>45.74</u>±1.23	27.7	0.20

Table 8: Test AUROC and AUPRC (%) under MNAR test masks on MIMIC-IV and eICU. Models are trained under MCAR. Best results are shown in bold and second-best results are underlined.

Model	MIMIC-IV				eICU			
	Readmission		Mortality		Readmission		Mortality	
	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC
Mean	74.44±0.24	30.40±0.54	80.39±2.81	24.63±3.34	77.93±0.20	48.00±0.16	90.35±0.06	44.18±0.15
SMIL	75.06±0.24	31.67±0.40	80.56±2.69	24.83±2.76	78.01±0.10	48.07±0.03	90.69±0.04	45.02±0.20
FeatProp	75.14±0.08	31.22±0.19	82.80±0.05	26.90±0.46	78.15±0.05	47.95±0.10	90.37±0.42	44.82±0.70
GRAPE	75.68±0.32	31.55±0.72	<u>83.80</u>±0.37	<u>28.65</u>±0.57	78.09±0.37	47.88±0.48	89.89±0.05	43.19±0.10
M3Care	74.48±0.16	30.40±0.31	81.93±0.14	25.38±0.46	<u>78.59</u>±0.05	48.70±0.17	89.88±0.74	43.90±1.20
MUSE	75.27±0.06	31.52±0.35	83.18±0.59	28.49±0.94	78.29±0.27	48.59±0.22	<u>90.93</u>±0.21	<u>45.72</u>±0.76
MUSE+	74.98±0.55	31.08±0.47	82.83±0.31	27.78±0.08	78.28±0.05	<u>48.71</u>±0.20	90.49±0.58	44.67±0.81
RAMCare	<u>75.75</u>±0.08	<u>31.90</u>±0.20	82.95±0.32	26.71±0.90	77.99±0.99	48.59±1.04	**91.36**±0.05	**47.07**±0.32
DiffPuter	75.00±0.13	30.81±0.51	82.14±0.37	26.25±0.35	78.46±0.01	48.40±0.11	90.80±0.11	45.41±0.24
MEDUSA	**76.08**±0.11	**32.10**±0.19	**84.02**±0.11	**29.74**±0.48	**78.60**±0.24	**48.89**±0.11	90.67±0.25	45.54±1.14

ties. For each mask, one modality is randomly selected as input, and each of the other modalities is dropped with a probability given by a logistic model of the input modality raw features. The input modality is then itself dropped completely at random with the same probability, so that the missingness of the other modalities depends on values that may be unobserved. As in the MCAR setting, we draw ten test masks, shared across all methods. Models are the same checkpoints trained under MCAR, so this experiment measures robustness to a change in the missingness mechanism at test time. Results are reported in Table 8.

Table 8 reports test AUROC and AUPRC under MNAR test masks on both tasks of MIMIC-IV and eICU, under the same protocol as in Section 4.1. The results mirror those obtained under MCAR (Table 1): MEDUSA achieves the best AUROC and AUPRC on both tasks of MIMIC-IV and on eICU readmission, while RAMCare performs better on eICU mortality. This shows that MEDUSA remains competitive when the missingness mechanism changes at test time.

D Computational Resources and Codebase

All our experiments were run on a single Linux machine equipped with one AMD EPYC 7502 32-Core CPU, a RAM with 500 GB, and one Tesla V100S-PCIE GPU with 32 GB. We release our codebase for reproducibility at this anonymous link: `https://anonymous.4open.science/r/MEDUSA-B415`.