GraphWalker: Graph-Guided In-Context Learning for Clinical Reasoning on Electronic Health Records

Anonymous ACL submission

Abstract

Clinical Reasoning on Electronic Health Records (EHRs) is a fundamental yet challenging task in modern healthcare. While in-context learning (ICL) offers a promising inference-time adaptation paradigm for large language models (LLMs) in EHR reasoning, existing methods face three fundamental challenges: (1) *Perspective Limitation*, where data-driven similarity fails to align with LLM reasoning needs and model-driven signals are constrained by limited clinical competence; (2) *Cohort Awareness*, as demonstrations are selected independently without modeling population-level structure; and (3) *Information Aggregation*, where redundancy and interaction effects among demonstrations are ignored, leading to diminishing marginal gains. To address these challenges, we propose *GraphWalker*, a principled demonstration selection framework for EHR-oriented ICL. *GraphWalker* (i) jointly models patient clinical information and LLM-estimated information gain by *integrating data-driven and model-driven perspectives*, (ii) incorporates *Cohort Discovery* to avoid noisy local optima, and (iii) employs a *Lazy Greedy Search with Frontier Expansion* algorithm to mitigate diminishing marginal returns in information aggregation. Extensive experiments on multiple real-world EHR benchmarks demonstrate that *GraphWalker* consistently outperforms state-of-the-art ICL baselines, yielding substantial improvements in clinical reasoning performance. Our code is open-sourced at https://anonymous.4open.science/status/GraphWalker-4473.

1 Introduction

Clinical Reasoning on Electronic Health Records (EHRs) is a fundamental yet challenging task in modern healthcare, underpinning critical applications (Ma et al., 2023; Xu et al., 2023a,b, 2024; Fang et al., 2023; Liao et al., 2025) such as diagnosis support, risk stratification, and treatment planning. With the rapid advancement of large language models (LLMs), an increasing body of work has explored leveraging LLMs for clinical reasoning over EHRs. Early efforts primarily relied on pre-training (Zhang et al., 2025a) or post-training (Ding et al., 2025a; Fang et al., 2025; Ding et al., 2025b) strategies to adapt LLMs to the clinical domain. However, empirical evidence (Brown et al., 2024; Chen et al., 2024) increasingly suggests that, due to their limited sensitivity to temporal structure, LLMs struggle to reliably activate clinically meaningful longitudinal patterns stored in parametric memory (Zhu et al., 2024a), even after domain adaptation. Inspired by clinical practice, where physicians often reason about a patient by recalling and comparing similar past cases (Ten Cate et al., 2017), in-context learning (ICL) (Brown et al., 2020) has emerged as a practical and effective alternative.

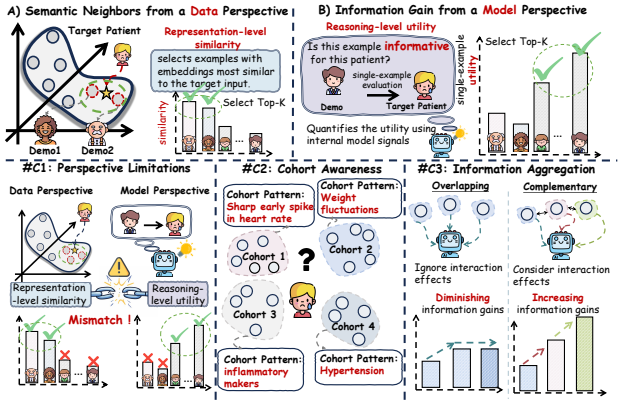

Figure 1: Illustration of existing ICL paradigms (top) and challenges (bottom) in EHR-based clinical reasoning.

Existing ICL-based literature can be broadly categorized into two paradigms.

- ***Semantic Neighbors*** *from a Data Perspective*: As illustrated in Fig. 1(A), this paradigm (Liu et al., 2022; Hongjin et al., 2022; Robertson et al., 2009) leverages a pre-trained representation model(e.g., SMART) (Yu et al., 2024) to embed instances into a latent space, and then selects examples by retrieving instances whose embeddings are most similar to that of the target input.
- ***Information Gain*** *from a Model Perspective*: As illustrated in Fig. 1(B), this paradigm (Gonen et al., 2023; Peng et al., 2024; Liu et al.; Li et al., 2025) quantifies the utility of candidate examples using internal model signals, such as conditional entropy reduction or gradient-based influence measures, and select the top-k examples that are expected to maximally improve the model's predictive confidence.

Through an extensive review of prior literature, we

conduct a systematic analysis of existing methods (Details see Section. 2) and identify **three core challenges** challenges shared by existing works, as shown in Fig. 1(C).

- *Perspective Limitation*: ***Semantic Neighbors*** implicitly assumes that embedding similarity aligns with the LLM's reasoning needs—an assumption that does not consistently hold in practice. Empirical observations show that highly similar patient trajectories may still contribute little to downstream predictions, suggesting a mismatch between representation-level similarity and reasoning-level utility. Although ***Information Gain*** is more directly aligned with the target model, these approaches are inherently constrained by the LLM's clinical competence; when the model lacks sufficient domain knowledge, its internal uncertainty signals may fail to reflect clinically meaningful longitudinal patterns.
- *Cohort Awareness*: ***Both paradigms*** focus on retrieving isolated similar patients, implicitly treating each example as independent. However, clinical reasoning is inherently population-driven. Selecting individual neighbors without modeling cohort structure can amplify noise from idiosyncratic cases and lead to unstable or biased reasoning contexts.
- *Information Aggregation*: ***Both paradigms*** typically assume that the utility of selected demonstrations accumulates linearly. In practice, however, multiple similar demonstrations often encode overlapping clinical signals, resulting in diminishing marginal gains when added to the context. Our analysis suggests that ignoring redundancy and interaction effects among demonstrations leads to inefficient use of the limited context window.

These challenges motivate the need for a principled demonstration selection framework that is both clinically-informed and model-aware. To this end, we propose *GraphWalker*, a novel demonstration selection algorithm designed to enhance LLM-based clinical reasoning on EHRs. *GraphWalker* explicitly *integrates data perspective and model perspective* by encoding patient clinical information using pre-trained EHR representation models to construct a patient semantic graph, while leveraging LLM-estimated information gain to guide traversal and selection. To capture cohort awareness, *GraphWalker* replaces instance-level retrieval with *Cohort Discovery*, enabling reasoning over clinically coherent patient groups rather than isolated exemplars. Finally, to address redundancy and interaction among demonstrations, we introduce a *Lazy Greedy Search with Frontier Expansion*, which formulates demonstration selection as a combinatorial optimization problem and efficiently approximates locally optimal demonstration sets under a constrained context budget.

To summarize, our contributions are threefold:

- **Insightfully**, we conducted an in-depth review of existing studies that leverage ICL to enhance LLM-based clinical reasoning on EHR. Through a series of experimental analyses and empirical investigations, we systematically identify and address three key challenges: *Perspective Limitation*, *Cohort Awareness*, and *Information Aggregation*.
- **Technically**, we design and implement GraphWalker, a three-stage demonstration selection algorithm: it (1) jointly models patient's clinical information and LLM's information gain by *integrating data-driven and model-driven perspectives*; (2) leverages *Cohort Discovery* to avoid being trapped in local optima caused by noisy nodes; and (3) introduces a *Lazy Greedy Search with Frontier Expansion* algorithm to address the diminishing marginal returns inherent in information aggregation.
- **Experimentally**, we conduct extensive evaluations on multiple real-world EHR benchmarks, demonstrating consistent and substantial improvements over state-of-the-art ICL baselines.

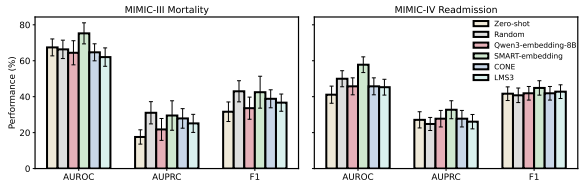

Figure 2: Analysis of the Limitations of a Single Perspective.

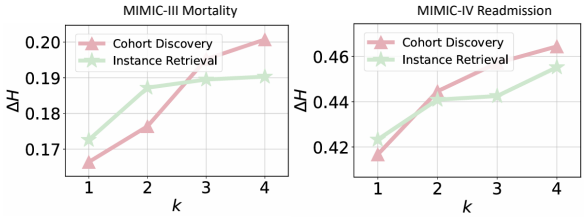

Figure 3: Analysis of Cohort Discovery *vs.* Instance Retrieval.

2 Observation & Motivation

In this section, we conduct an in-depth analysis of existing demonstration selection algorithms and present key observations obtained from our preliminary investigations, which motivate the algorithmic design of *GraphWalker*. Detailed experimental settings can be found in the Appendix. E

2.1 Motivation on Perspective Limitation

We introduce the following baseline methods: Zero-shot (no demonstration selection), Random selection, two **Semantic Neighbor** approaches—Qwen3-Embedding-8B (Yang et al., 2025) and SMART (Yu et al., 2024) (a model pre-trained on a large-scale EHR corpus)—and two **Information Gain** methods, CONE (Peng et al., 2024) and LMS3 (Liu et al.). The experimental results are shown in Figure. 2. The results indicate that SMART achieves the best performance in most cases, confirming the importance of clinical knowledge in demonstration selection. Moreover, we observe that, in the majority of cases, methods based on CONE and LMS3 consistently outperform Qwen3-Eembedding-8B. This suggests that,

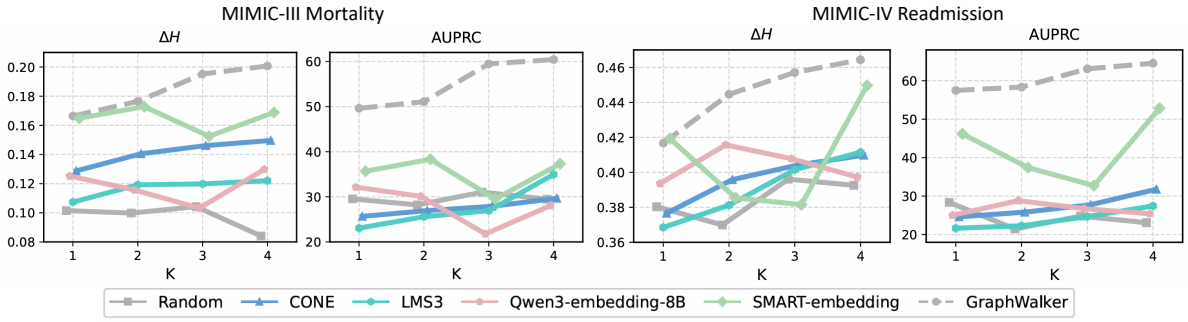

Figure 4: Analysis of Information Aggregation of Multiple Demonstrations using various Demonstration Selection Algorithm.

even in the absence of explicit clinical knowledge, information gain serves as an effective criterion for demonstration selection. Overall, these findings lead to our first key observation:

Observation I: *Both clinical knowledge (Data Perspective) and information gain (Model Perspective) provide valuable guidance for demonstration selection.*

2.2 Motivation on Cohort Awareness

We further analyze traditional instance retrieval and our proposed cohort discovery strategy. Specifically, we use SMART as the representation encoder and construct a graph over patients. Traditional instance retrieval selects the most relevant instance solely based on embedding similarity and then performs local walks around this instance to select the remaining demonstrations. In contrast, our proposed cohort discovery approach identifies the most similar cohort and selects demonstrations from within that cohort.

The experimental results are shown in Figure. 3. The results indicate that instance retrieval plays a critical role when the number of demonstrations is small; however, as the number of demonstrations increases, instance retrieval becomes less effective than cohort discovery. We attribute this to the following reasons: although instance retrieval can identify a single highly similar demonstration, this instance may be an outlier or belong to a small, sparse cluster, which can adversely affect subsequent demonstration selection and lead to a local optimum. In contrast, cohort discovery accounts for the relational structure among patients and provides a more robust basis for selecting subsequent demonstrations. Based on this analysis, we derive our second key observation:

Observation II: *Instance retrieval strategies tend to overemphasize a single most similar instance during the initial demonstration selection, which can lead to local optima; as the number of demonstrations increases, their performance degrades relative to cohort discovery approaches.*

2.3 Motivation on Information Aggregation

We further analyze the performance of existing algorithms in the multi-demonstration setting. In this experiment, we additionally introduce the LLM's conditional entropy reduction (ΔH) as an analysis metric(see details in Appendix. E). Beyond the results shown in Figure. 4, we observe the following findings: (1) The model performance closely follows the trend of ΔH, indicating that ΔH can serve as an effective guiding signal for demonstration selection. This observation further strengthens our motivation to integrate both data-driven and model-driven perspectives. (2) **Semantic Neighbor** methods completely overlook the issue of information aggregation; in multi-demonstration settings, their performance degrades due to information redundancy or conflicts. In contrast, although **Information Gain** methods maintain a monotonic performance improvement as the number of demonstrations increases, they exhibit pronounced diminishing marginal returns due to insufficient consideration of the information set as a whole. Here, we derive our third key observation:

Observation III: *Both* ***Semantic Neighbor*** *and* ***Information Gain*** *approaches insufficiently account for Information Aggregation, leading to performance degradation or diminishing marginal returns in multi-demonstration settings.*

Summary. Based on the above observations, we propose *GraphWalker*, which addresses three key challenges in demonstration selection through the design of point-to-point strategies. Specifically, *GraphWalker* (1) *integrates data perspective and model perspective* to aggregate clinical knowledge and leverage information gain as a guiding signal (for ***Observation I***), (2) replaces *Instance Retrieval with Cohort Discovery*, enabling reasoning over clinically coherent patient groups rather than isolated exemplars (for ***Observation II***), (3) introduces a *Lazy Greedy Search with Frontier Expansion*, which formulates demonstration selection as a combinatorial optimization problem to address Information Aggregation (for ***Observation III***).

3 Methodology

As illustrated in Figure 5, *GraphWalker* includes three stages:

- **# Stage 1: Patient Graph Construction and Cohort Discovery**. builds a population-level patient graph and discovers coherent cohorts, enabling the capture of shared clinical patterns beyond isolated cases.
- **# Stage 2: Instance-aware Cohort Retrieval and Anchor Initialization**. retrieves clinically relevant cohorts for target patient and initializes anchor nodes,

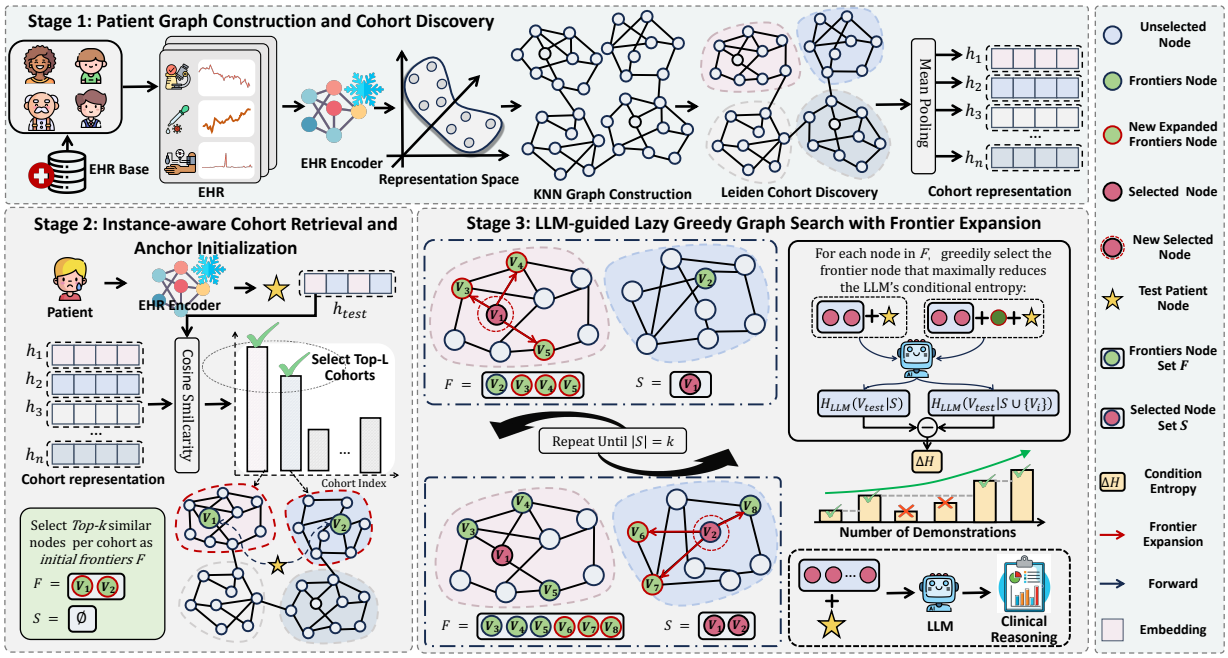

Figure 5: Illustration of *GraphWalker*.

providing a reliable and informative candidate set.

- **# Stage 3: LLM-guided Lazy Greedy Search with Frontier Expansion**. iteratively searches over the cohort subgraphs and constructs the demonstration set by maximizing LLM-guided joint information gain.

3.1 Patient Graph Construction and Cohort Discovery

This stage aims to capture shared clinical patterns beyond isolated patient cases, providing a structured prior over patient clinical similarity to guide downstream reasoning. To this end, we perform *Patient Representation* using a pretrained EHR model, construct a similarity-based patient graph via *Population-level Graph Construction*, and apply *Graph-based Cohort Discovery* to capture shared clinical patterns.

Patient Representation. Due to the heterogeneous and long-horizon nature of EHR data, semantic embeddings are insufficient for modeling clinically meaningful patient similarity. We therefore adopt a pretrained Transformer-based EHR encoder (e.g., *SMART* (Yu et al., 2024)) to obtain representations that capture longitudinal clinical patterns. Formally, a patient's EHR record is represented as a sequence of T time-ordered visits $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T]$. Given a collection of patient records from the EHR base (i.e., the training set) $\{\mathbf{X}_i\}_{i=1}^N$, a pretrained encoder $\mathcal{M}_{\text{exp}}$ maps each longitudinal record to a fixed-dimensional embedding:

$$\{\mathbf{h}_i\}_{i=1}^N = \mathcal{M}_{\text{exp}}\left(\{\mathbf{X}_i\}_{i=1}^N\right), \qquad \mathbf{h}_i \in \mathbb{R}^d. \quad (1)$$

which serves as the patient representation for subsequent graph construction.

Population-level Graph Construction. To capture clinically meaningful relationships among patients beyond isolated representations, we construct a population-level patient graph that explicitly encodes local similarity structure in the representation space. Formally, given patient embeddings $\{\mathbf{h}_i\}_{i=1}^N$, we define a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where each node $v_i \in \mathcal{V}$ corresponds to a patient. An edge $(i, j) \in \mathcal{E}$ is established if patient j is among the k_g nearest neighbors of patient i under cosine similarity:

$$\mathcal{E} = \{(i, j) \mid \text{rank}_j(\text{sim}(\mathbf{h}_i, \mathbf{h}_j)) \leq k_g\}, \quad (2)$$

where $\text{sim}(\cdot, \cdot)$ denotes cosine similarity.

Graph-based Cohort Discovery. To move beyond isolated patient cases and capture shared clinical patterns at the cohort level, we further *learn a partition* of the patient graph into coherent cohorts of similar patients. Specifically, we apply the *Leiden* (Traag et al., 2019) algorithm to *learn a partition* of the patient graph into a set of cohort subgraphs $\{\mathcal{C}_m\}_{m=1}^M$ by maximizing graph modularity $\mathcal{Q}$:

$$\mathcal{Q} = \frac{1}{2|\mathcal{E}|} \sum_{i,j} \left(A_{ij} - \frac{d_i d_j}{2|\mathcal{E}|}\right) \mathbb{I}(c_i = c_j), \quad (3)$$

where A_{ij} denotes the adjacency matrix of $\mathcal{G}$, d_i is the degree of node i, and $\mathbb{I}(\cdot)$ indicates whether two nodes belong to the same cohort. As a result, each cohort subgraph exhibits dense internal connectivity, while connections between different cohort subgraphs are relatively sparse. For each cohort $\mathcal{C}_m$, we compute a cohort prototype by mean pooling patient embeddings:

$$\mathbf{z}_m = \frac{1}{|\mathcal{C}_m|} \sum_{i \in \mathcal{C}_m} \mathbf{h}_i. \quad (4)$$

These prototypes summarize cohort-level clinical patterns and serve as compact cohort representations for subsequent instance-aware retrieval.

3.2 Instance-aware Cohort Retrieval and Anchor Initialization

This stage aims to locate the target patient within clinically coherent population-level contexts and establish principled starting points for subsequent demonstration search. Specifically, we perform *Instance-aware Cohort Retrieval* by aligning the target patient representation with cohort centroids that capture shared clinical patterns, followed by *Anchor Initialization* within the selected cohorts to define a focused search frontier.

Instance-aware Cohort Retrieval. Given a target patient with EHR record $\mathbf{X}^{(q)}$, we obtain its embedding $\mathbf{h}^{(q)} = \mathcal{M}_{\text{exp}}(\mathbf{X}^{(q)})$ using the same pretrained EHR encoder as in Stage 1. To account for cohort awareness and avoid reasoning over isolated patient instances, we retrieve clinically relevant cohorts by aligning $\mathbf{h}^{(q)}$ with cohort centroids, thereby grounding the subsequent demonstration search in population-level shared clinical patterns. Specifically, we compute the similarity between $\mathbf{h}^{(q)}$ and each cohort centroid $\mathbf{z}_m$ (Eq. (4)) and select the top-K_c cohorts:

$$\mathcal{C}_{\text{ret}}^{(q)} = \text{Top-}K_c\big(\text{sim}(\mathbf{h}^{(q)}, \mathbf{z}_m)\big), \tag{5}$$

where $\text{sim}(\cdot,\cdot)$ denotes cosine similarity. This step situates the test patient within clinically coherent regions of the patient graph, providing a focused context for subsequent anchor initialization.

Anchor Initialization. Given the retrieved cohorts $\mathcal{C}_{\text{ret}}^{(q)}$, we initialize the search frontier by selecting anchor nodes within each cohort. For each cohort $\mathcal{C}_m \in \mathcal{C}_{\text{ret}}^{(q)}$, we select the top-$K_a$ nodes whose embeddings are most similar to the target patient embedding $\mathbf{h}^{(q)}$:

$$\mathcal{V}_m^{(q)} = \text{Top-}K_a\big(\text{sim}(\mathbf{h}^{(q)}, \mathbf{h}_i)\big), \quad v_i \in \mathcal{C}_m, \tag{6}$$

where $\text{sim}(\cdot,\cdot)$ denotes cosine similarity. The initial search frontier is defined as

$$\mathcal{F}^{(q)} = \bigcup_{\mathcal{C}_m \in \mathcal{C}_{\text{ret}}^{(q)}} \mathcal{V}_m^{(q)}. \tag{7}$$

3.3 LLM-guided Lazy Greedy Search with Frontier Expansion

This stage formulates demonstration selection as a combinatorial optimization problem and aims to iteratively construct a demonstration set that maximizes reasoning utility under a constrained context budget. It jointly leverages population-level clinical patterns encoded by the patient graph and LLM-guided evaluation of reasoning utility to efficiently approximate locally optimal demonstration sets. To achieve this, we alternate between *LLM-guided Greedy Selection*, which identifies the most informative example conditioned on the current demonstration composition, and *Graph-based Frontier Expansion*, which progressively exposes clinically related candidates for subsequent selection.

LLM-guided Greedy Selection. At each iteration, given the current frontier $\mathcal{F}^{(q)}$ and a partially constructed demonstration set $\mathcal{S}^{(q)}$, we select the next demonstration that maximizes the marginal *information gain* on the target case, conditioned on the current demonstration composition. Following prior work (Peng et al., 2024), we quantify information gain using an entropy-based formulation and extend it to a *composition-aware* criterion that captures the joint effect of multiple demonstrations.

Formally, let $H_\theta(x_{\text{test}} \mid \mathcal{S})$ denote the conditional entropy of the model's prediction on the test query x_{test} given a demonstration composition $\mathcal{S}$, where x_{test} is an unlabeled query input rather than a supervised target. This conditional entropy characterizes the model's degree of understanding of the target query under the given demonstration composition. It can be written in a difference form as

$$H_\theta(x_{\text{test}} \mid \mathcal{S}) = H_\theta(x_{\text{test}}, \mathcal{S}) - H_\theta(\mathcal{S}), \tag{8}$$

where $H_\theta(x_{\text{test}}, \mathcal{S})$ and $H_\theta(\mathcal{S})$ denote the cross-entropy of the full prompt and that of the demonstration composition alone, respectively. For a candidate node $v_i \in \mathcal{F}^{(q)}$, we define its *marginal information gain* with respect to the current demonstration composition $\mathcal{S}^{(q)}$ as

$$\begin{aligned}\Delta H_\theta(v_i \mid \mathcal{S}^{(q)}) = H_\theta(x_{\text{test}} \mid \mathcal{S}^{(q)}) \\ - H_\theta(x_{\text{test}} \mid \mathcal{S}^{(q)} \cup \{v_i\}).\end{aligned} \tag{9}$$

A larger $\Delta H_\theta(\cdot)$ indicates that v_i provides more complementary information that enhances the collective reasoning capacity of the current demonstration composition.

Within each iteration, we select the frontier node that maximizes the marginal information gain with respect to the current demonstration set:

$$v^* = \arg\max_{v_i \in \mathcal{F}^{(q)}} \Delta H_\theta\big(v_i \mid \mathcal{S}^{(q)}\big). \tag{10}$$

The selected node is then added to the demonstration set, updating the current demonstration composition:

$$\mathcal{S}^{(q)} \leftarrow \mathcal{S}^{(q)} \cup \{v^*\}. \tag{11}$$

For computational efficiency, we implement this procedure using a *lazy greedy* strategy, which maintains a priority queue over frontier nodes with cached marginal information gains and recomputes them only when necessary as $\mathcal{S}^{(q)}$ evolves. Further details of this lazy greedy strategy are provided in Appendix I. The updated demonstration composition subsequently triggers the graph-based frontier expansion step.

Graph-based Frontier Expansion. After each greedy selection step, we expand the search frontier to introduce new candidates that are conditionally relevant to the current demonstration composition. Given a newly selected node v^*, we update the frontier by adding its neighbors in the patient graph:

$$\mathcal{F}^{(q)} \leftarrow \mathcal{F}^{(q)} \cup \mathcal{N}(v^*), \tag{12}$$

where $\mathcal{N}(v^*) = \{v_j \mid (v^*, v_j) \in \mathcal{E}\}$ denotes the neighbors of v^*. This expansion exploits the local connectivity of the patient graph, where neighboring nodes

are expected to share locally coherent clinical patterns. Conditioned on the fact that v^* yields high information gain for the target case, its neighbors are more likely to provide complementary clinical evidence that further enhances the reasoning utility of the current demonstration composition. Previously selected nodes are excluded from the frontier to avoid redundancy.

Stopping Criterion. By alternating between LLM-guided greedy selection and graph-based frontier expansion, the algorithm incrementally explores clinically related regions of the patient graph while continuously optimizing the reasoning utility of the demonstration composition. The procedure terminates when either (i) a predefined budget of K demonstrations is reached, or (ii) no candidate in the current frontier yields a positive marginal information gain, thereby preventing the inclusion of misleading or uninformative demonstrations.

LLM Inference with Selected Demonstrations. After selection, the demonstration set $\mathcal{S}^{(q)}$ is used to construct a few-shot prompt(see Appendix J for details), which is then used by the target LLM to generate the final prediction. The full algorithm is provided in Appendix H.

4 Experiment

Datasets and Tasks. We evaluate our method on two widely used, publicly available EHR benchmarks, *MIMIC-III* (Johnson et al., 2016) and *MIMIC-IV* (Johnson et al., 2023), across standard clinical prediction tasks including in-hospital mortality, length-of-stay (LOS), and ICU readmission prediction. Following established EHR benchmarks (Gao et al., 2024; Zhu et al., 2024b), we adopt standard patient-level preprocessing and visit sequencing protocols. We include patients with at least two visits and use the last visit for prediction. Stratified train/validation/test splits are employed. Detailed dataset statistics, task definitions, and preprocessing procedures are provided in Appendix B and C.

Baselines. We compare *GraphWalker* with representative baselines spanning multiple paradigms:

- **Vanilla ICL baselines.** *Zero-shot* is a special case of ICL where no demonstration example is provided. *Random* randomly samples a fixed number of examples as demonstrations.
- **Supervised.** *Supervised fine-tuning(SFT)* adapts LLMs to downstream tasks by updating model parameters on task-specific labeled data.
- **Data Perspective Selection.** *Semantic-emb* and *EHR-emb* select demonstrations as *semantic neighbors* of the test query by performing top-k nearest-neighbor retrieval in embedding space, using *Qwen3-Embedding-8B* (Zhang et al., 2025b) (a general-purpose semantic encoder) and *SMART* (Yu et al., 2024) (an EHR-specific encoder), respectively.
- **Model Perspective Selection.** *SPELL* (Gonen et al., 2023) selects demonstrations by calculating their perplexity. *Influence* (Nguyen and Wong, 2023) estimates example importance via validation performance differences. *IDS* (Qin et al., 2024) selects demonstrations based on reasoning-path similarity. *CONE* (Peng et al., 2024) selects examples by minimizing the conditional entropy of the test input on a per-example basis, ignoring the complementarity among selected demonstrations. *LMS3* (Liu et al.) leverages LLM embedding similarity and gradient stability. *Delta-KNN* (Li et al., 2025) selects demonstrations by prioritizing examples that were empirically observed to be more helpful in past predictions.

See Appendix D for baselines implementation details.

Backbone Models. To evaluate the generality of our method, we instantiate it on multiple LLM backbones, including *Qwen3-32B* and *14B* (Yang et al., 2025), as well as *LLaMA-3.1-8B-Instruct* (Dubey et al., 2024).

Evaluation Metrics. For *mortality* and *readmission* prediction, we employ three widely used evaluation metrics, including the area under the receiver operating characteristic curve (AUROC), the area under the precision–recall curve (AUPRC), and the F1-score. For LOS prediction, following Harutyunyan et al. (Harutyunyan et al., 2019), we discretize LOS labels into multiple bins and evaluate performance using macro-averaged ROC (ma-ROC) and micro-averaged ROC (mi-ROC).

4.1 Experimental Results

Performance Comparison. As shown in Table 1, ***GraphWalker* consistently achieves the strongest and most stable performance across backbones and tasks**, confirming its effectiveness in EHR reasoning. Concretely, Compared to the second-best performance, *GraphWalker* yield an average relative improvement of 9.65%↑ in AUROC, 12.98%↑ in AUPRC, 7.89%↑ in F1, 3.35%↑ in ma-ROC and 0.73%↑ in mi-ROC across all settings, confirming its strong performance. Notably, *GraphWalker* consistently outperforms existing ***single-perspective*** methods. This result empirically validates the advantage of a **dual-view demonstration selection paradigm that bridges clinical similarity and LLM-internal reasoning dynamics**. Furthermore, in contrast to SFT, *GraphWalker* attains comparable or superior performance without parameter updates, underscoring its lightweight and efficient inference-time adaptation for EHR reasoning under data-scarce conditions.

Ablation Study. To examine the contribution of each component in *GraphWalker*, we conduct ablation experiments on two EHR benchmarks using *Qwen3-14B*. The experimental results were presented in Table 2. In *w/o EHR-emb*, replacing the EHR encoder with a semantic embedding model (*Qwen3-Embedding-8B*) results in a pronounced performance drop(e.g., an average 18.63%↓ F1), indicating that generic semantic similarity fails to capture clinically meaningful patient relationships required for effective graph-based search. In *w/o Cohort Discovery*, we do not partition patients into clinically coherent cohorts, but instead directly initialize anchors by selecting the top-K_a nearest patients from the graph, which leads to a noticeable performance degradation(e.g., an average 6.09%↓ AUPRC). This result high-

Method		MIMIC-III Mortality			MIMIC-III LOS		MIMIC-IV Readmission		
Paradigm	**Approach**	AUROC ↑	AUPRC ↑	F1 ↑	ma-ROC ↑	mi-ROC ↑	AUROC ↑	AUPRC ↑	F1 ↑
					Qwen3-32B				
Vanilla	Zero-shot	65.67±5.75	30.54±7.56	38.43±6.99	55.79±3.33	23.53±2.46	44.19±4.57	25.06±3.88	42.57±3.83
	Random	59.90±5.06	24.63±5.29	36.22±4.94	59.50±3.51	80.44±2.00	48.25±4.65	28.13±4.51	42.22±3.77
Supervised	SFT	51.34±1.36	19.58±3.50	29.50±3.96	60.47±3.68	40.20±2.69	52.45±3.65	26.11±3.56	40.20±3.93
Data Perspective	Semantic-emb	60.62±5.80	19.87±4.61	32.08±5.59	57.65±3.65	74.16±2.38	47.86±4.71	26.07±3.89	42.47±3.85
	EHR-emb	85.02±3.21	58.56±8.02	58.84±6.01	63.02±3.33	67.00±2.71	61.32±4.45	44.25±5.92	46.24±4.06
Model Perspective	SPELL	63.26±4.98	27.26±5.64	38.33±5.45	53.87±3.76	75.06±2.27	48.56±4.77	25.90±4.11	40.17±3.86
	Influence	62.45±5.55	29.28±6.41	40.18±6.30	54.24±3.02	41.07±1.56	50.45±3.49	27.63±3.42	42.69±3.94
	IDS	56.72±5.64	28.37±6.91	35.23±5.03	52.47±3.12	45.82±2.47	46.93±4.66	27.71±4.34	43.16±3.79
	CONE	64.28±6.59	27.83±7.35	36.05±6.90	56.30±3.73	68.90±2.64	45.35±4.83	27.85±4.77	41.73±3.79
	LMS3	62.03±5.12	25.14±5.03	36.68±4.79	58.14±3.73	73.15±2.55	45.28±4.40	26.09±4.01	42.79±3.82
	Delta-KNN	61.08±5.83	29.41±6.34	39.36±5.92	56.55±4.09	64.86±2.64	51.75±4.68	26.74±4.26	40.37±3.90
Ours	*Graph-Walker*	**85.88±3.16**	**59.19±8.04**	**61.89±6.07**	**67.40±3.51**	**83.60±2.34**	**77.57±4.01**	**63.09±6.27**	**59.51±4.46**
	Impro(%)	*+0.86*	*+0.63*	*+3.05*	*+4.38*	*+3.16*	*+16.25*	*+18.84*	*+13.27*
					Qwen3-14B				
Vanilla	Zero-shot	64.09±5.26	17.58±3.98	31.62±5.44	65.22±3.00	37.49±2.48	41.11±4.77	27.11±4.51	41.66±3.80
	Random	66.33±5.12	31.04±6.89	43.00±5.88	58.12±1.98	61.28±3.35	49.98±4.47	24.81±3.63	40.77±4.05
Supervised	SFT	68.34±6.29	17.48±4.88	30.43±6.35	65.31±3.80	**86.51±1.85**	53.23±2.61	27.44±3.91	39.36±3.87
Data Perspective	Semantic-emb	64.43±6.71	21.78±6.09	33.60±6.16	57.06±3.77	66.60±2.60	45.76±4.72	27.75±4.60	41.89±3.79
	EHR-emb	75.24±5.86	29.52±8.20	42.49±8.88	62.03±3.13	66.98±2.53	57.79±4.37	32.73±5.01	44.87±4.01
Model Perspective	SPELL	75.37±4.83	32.23±6.55	50.32±6.86	63.73±3.65	75.78±2.18	48.82±4.37	24.37±3.38	40.03±3.98
	Influence	60.54±5.64	27.20±6.59	36.10±6.88	50.03±2.64	62.70±2.30	52.85±3.28	28.55±3.75	43.19±3.85
	IDS	59.14±4.53	19.73±3.93	32.97±4.45	49.86±2.06	37.41±1.11	47.41±4.51	25.43±3.48	42.68±3.79
	CONE	64.69±4.74	27.90±5.44	38.82±5.01	57.84±3.66	67.91±2.57	45.76±4.72	27.75±4.60	41.89±3.79
	LMS3	66.54±5.61	26.92±6.55	39.07±7.46	59.44±3.21	31.89±2.54	44.34±4.57	24.68±3.47	42.28±3.81
	Delta-KNN	51.52±6.34	15.66±4.31	25.22±4.87	59.04±3.61	63.43±2.51	47.11±4.81	24.41±3.90	40.00±3.91
Ours	*Graph-Walker*	**84.19±3.35**	**59.46±7.64**	**58.24±5.89**	**69.51±3.16**	84.64±2.14	**75.91±4.11**	**60.61±6.19**	**58.01±5.33**
	Impro(%)	*+8.82*	*+27.23*	*+7.92*	*+4.20*	*-1.87*	*+18.12*	*+27.88*	*+13.14*
					Llama3.1-8B-Instruct				
Vanilla	Zero-shot	65.35±4.90	21.72±4.61	35.75±5.22	43.16±4.25	74.13±2.63	54.40±5.02	29.88±4.99	41.24±4.14
	Random	52.04±5.55	18.81±4.78	28.90±4.11	45.66±4.24	80.14±2.31	48.63±5.03	27.06±4.68	39.85±3.85
Supervised	SFT	67.41±5.36	26.41±6.11	35.73±5.94	52.60±3.86	81.49±2.26	45.37±3.41	22.94±3.22	37.25±4.00
Data Perspective	Semantic-emb	56.90±5.44	22.32±4.69	35.15±5.10	53.78±3.87	81.49±2.26	43.55±4.33	23.69±3.25	42.37±3.83
	EHR-emb	65.78±4.24	24.30±4.39	41.89±5.14	45.72±4.08	74.78±2.63	63.75±4.78	46.32±6.37	49.81±4.88
Model Perspective	SPELL	55.43±4.83	20.92±4.44	34.51±4.45	49.48±4.07	82.02±2.24	48.11±4.67	25.03±4.09	40.09±3.86
	Influence	61.73±5.31	31.80±6.55	38.39±5.19	55.57±4.39	82.63±2.37	49.64±3.99	28.40±4.17	42.65±3.81
	IDS	59.89±5.52	28.40±6.61	36.11±5.13	42.12±3.58	71.14±2.10	45.25±4.52	24.86±3.59	42.65±3.81
	CONE	50.40±5.29	18.93±3.97	32.79±4.54	38.31±3.82	76.94±2.53	51.67±4.73	29.42±4.45	42.73±3.86
	LMS3	51.80±5.73	16.41±3.38	29.37±4.67	39.60±3.86	74.05±2.68	54.77±4.81	30.95±4.55	45.41±4.73
	Delta-KNN	57.42±4.91	21.48±4.12	33.61±4.43	47.69±3.82	75.71±2.60	51.23±4.62	26.50±4.15	41.13±4.00
Ours	*Graph-Walker*	**76.71±3.58**	**35.09±6.40**	**50.13±5.52**	**57.06±3.98**	**83.55±2.16**	**68.32±4.19**	**46.37±6.65**	**51.54±4.45**
	Impro(%)	*+9.30*	*+3.29*	*+8.24*	*+1.49*	*+0.92*	*+4.57*	*+0.05*	*+1.73*

Table 1: Performance comparison on MIMIC-III and MIMIC-IV datasets. All evaluation metrics are defined such that higher values indicate better performance (↑). The best results are highlighted in **Bold**, while the second-best results are underline. The *Impro(%)* indicates the relative improvement of *GraphWalker* over the second-best performance.

Method	MIMIC-III Mortality			MIMIC-III LOS		MIMIC-IV Readmission		
	AUROC ↑	AUPRC ↑	F1 ↑	ma-ROC ↑	mi-ROC ↑	AUROC ↑	AUPRC ↑	F1 ↑
GraphWalker	**84.19±3.35**	**59.46±7.64**	**58.24±5.89**	**69.51±3.16**	**84.64±2.14**	**75.91±4.11**	**60.61±6.19**	**58.01±5.33**
w/o EHR-emb	61.09±5.32	31.13±6.81	37.50±5.73	62.03±3.40	72.63±2.46	45.30±4.55	26.69±4.20	41.74±3.79
w/o Cohort	81.19±3.78	48.27±8.01	55.06±6.18	65.85±3.75	81.16±2.27	75.65±4.04	59.61±6.09	56.87±5.19
w/o Greedy & Frontier Expansion	69.43±5.80	46.72±8.24	55.34±7.22	60.93±3.30	53.33±2.52	62.43±4.72	43.69±6.25	47.24±4.90
w/o Early Stop	81.34±3.74	48.66±7.91	55.99±6.18	66.31±3.23	82.77±2.14	75.53±4.09	59.13±6.12	57.38±5.22

Table 2: Ablation study of *GraphWalker* using *Qwen3-14B* as backbone LLM on MIMIC-III and MIMIC-IV datasets. All evaluation metrics are defined such that higher values indicate better performance (↑). The best results are highlighted in **Bold**.

lights the importance of **cohort awareness**: replacing instance-level retrieval with *Cohort Discovery* enables reasoning over clinically coherent patient groups, reducing noise from idiosyncratic cases and yielding more stable demonstration contexts.

We further examine the role of greedy search with frontier expansion. In *w/o Greedy & Frontier Expansion*, demonstrations are selected solely based on their individual ΔH scores within retrieved cohorts, implicitly assuming linear accumulation of utility. This results in a clear performance drop (e.g., an average 14.83%↓ AUPRC), indicating that independently scoring demon-

Method	MIMIC-III Mortality			MIMIC-IV Readmission		
	AUROC↑	AUPRC↑	F1↑	AUROC↑	AUPRC↑	F1↑
SMART	80.64	45.40	47.18	74.41	54.62	57.38
+Ours	**84.19**	**59.46**	**58.24**	**75.91**	**60.61**	**58.01**
Impro(%)	+3.55	+14.06	+11.06	+1.50	+5.99	+0.63
ConCare	69.19	36.01	40.65	70.30	41.87	50.26
+Ours	**70.46**	**36.93**	**45.90**	**77.01**	**59.49**	**59.44**
Impro(%)	+1.27	+0.92	+5.25	+6.71	+17.62	+9.68
AdaCare	60.67	19.37	26.81	78.40	59.37	54.87
+Ours	**65.61**	**27.13**	**37.89**	**80.45**	**62.72**	**60.04**
Impro(%)	+4.94	+7.76	+11.08	+2.05	+2.99	+5.17

Table 3: Performance of *GraphWalker* with different EHR encoders on MIMIC-III and MIMIC-IV, using *Qwen3-14B* as backbone LLM. Relative improvements (%) are reported w.r.t. the corresponding encoder baselines.

Paradigm	Method	CMB	MedQA	Avg
Vanilla ICL	Zero-shot	80.21±0.92	56.56±3.59	68.39
	Random	81.50±2.78	52.96±3.53	67.23
Data Perspective	Semantic-emb	80.52±2.70	59.58±3.52	70.05
Model Perspective	SPELL	81.98±2.73	54.10±3.49	68.04
	Influence	83.50±2.64	53.02±3.50	68.26
	IDS	79.57±2.81	52.59±3.57	66.08
	CONE	80.41±2.82	59.17±3.51	69.79
	LMS3	82.02±2.78	54.09±3.60	68.06
	Delta-KNN	83.02±2.66	53.47±3.42	68.25
Ours	**Graph-Walker**	**84.05±2.58**	**61.10±3.46**	**72.58**

Table 4: Performance comparison on MedQA and CMB benchmarks using *Qwen3-14B* as the backbone LLM. **Avg** denotes the arithmetic mean across tasks without variance. Higher values indicate better performance (↑).

strations fails to account for redundancy and interaction effects, leading to diminishing marginal gains. Moreover, in *w/o Early Stop*, removing the early stopping criterion allows demonstrations with negative marginal gains to be included, which further degrades performance. Together, these results highlight that effective demonstration selection requires explicitly modeling redundancy and interaction effects to efficiently utilize a constrained context budget.

Effect of Different EHR Encoders. To assess the robustness of *GraphWalker* to the choice of EHR representation, we evaluate it with multiple pretrained EHR encoders of different architectures and capacities, including SMART, ConCare (Ma et al., 2020b), and AdaCare (Ma et al., 2020a). Each EHR encoder is also fine-tuned as a task-specific predictor on the corresponding downstream task, and its performance is reported as the encoder baseline. As shown in Table 3, *GraphWalker* consistently improves performance across all encoders and tasks, indicating that its effectiveness is not tied to any specific EHR representation. These results suggest that the gains primarily stem from the dual-view demonstration selection mechanism, rather than from a particular encoder design.

Information Dynamics and Shot Scaling Analysis. Figure 4 shows that *GraphWalker* uniquely achieves a stable increase in ΔH with more demonstrations. By optimizing joint information gain at the composition level rather than scoring examples independently, it avoids misleading samples and alleviates information saturation under shot scaling. Moreover, as shown in Figure 6 and Figure 8, *GraphWalker* consistently achieves the best performance as the number of demonstrations increases when compared with a broader set of baselines. Additional shot-scaling results and detailed discussion are provided in Appendix K.

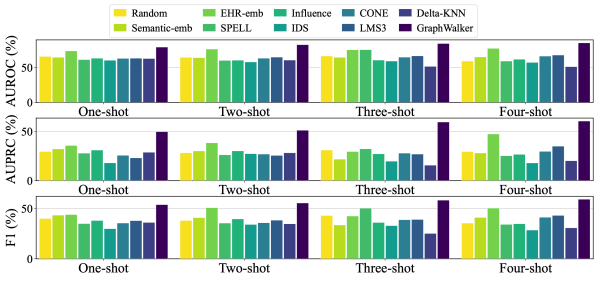

Figure 6: Shot scaling performance on MIMIC-III Mortality with Qwen3-14B as the backbone LLM.

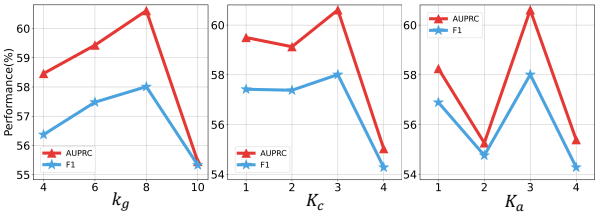

Figure 7: Performance of *GraphWalker* over different hyperparameters on MIMIC-IV.

Extended Investigations and Key Insights. We further extend our evaluation to **Additional Text-based Medical Benchmarks**. As shown in Table 4, *GraphWalker* consistently improves performance on these benchmarks, achieving an average gain of 3.6%↑ (see Appendix L for details), highlighting its potential for broader real-world clinical applications. In addition, we conduct a **Hyper-Parameter Analysis** of *GraphWalker* to support future applications and practical deployment. The results are summarized in Figure 7, with detailed analysis provided in Appendix M. We also report a runtime analysis in Appendix N, which shows that although *GraphWalker* incurs additional computation compared to baseline selection strategies, the overhead is limited and acceptable under typical clinical deployment settings. Finally, we provide a **Case Study** in Appendix O to further illustrate the advantages of *GraphWalker* in clinical reasoning.

5 Conclusion

In this work, we propose *GraphWalker*, a dual-view demonstration selection framework that bridges clinically grounded similarity modeling with LLM-internal reasoning dynamics, while explicitly incorporating cohort awareness and interaction-aware information aggregation. Extensive experiments demonstrate that *GraphWalker* consistently improves clinical reasoning performance across diverse benchmarks and settings. We advocate a broader perspective that principled, interaction-aware demonstration selection is essential for reliable and clinically aligned ICL with LLMs.

6 Limitations

While *GraphWalker* effectively addresses the limitations of single-perspective demonstration selection in EHR-based clinical reasoning, it has not yet been evaluated on substantially larger LLMs due to limited computational resources. In this work, we conduct experiments on LLMs with up to 8B and 14B parameters; extending the evaluation to larger-scale models would further strengthen the empirical evidence and help assess the scalability of our approach. In addition, *GraphWalker* introduces extra inference-time computation for demonstration selection, which may become non-negligible for extremely large LLMs or latency-sensitive clinical applications. Future work may explore more efficient exploration and pruning strategies to reduce selection overhead while preserving the benefits of cohort-aware and interaction-aware demonstration selection.

7 Ethical Considerations

Clinical decision-making is a complex and high-stakes process that relies on comprehensive medical evidence and professional judgment. Although LLMs can assist clinicians by organizing information, highlighting relevant patterns, or supporting exploratory reasoning, there is a risk that their outputs may be misunderstood or inappropriately treated as definitive medical advice. We emphasize that the system studied in this work is not intended to function as a diagnostic tool, nor to replace qualified healthcare professionals. Any model-generated reasoning or suggestions should be interpreted only as auxiliary information and must be considered alongside established clinical records, examinations, and expert assessment. All experiments in this study are conducted on publicly available and de-identified datasets, and no personally identifiable information or human subject data is involved. We acknowledge that the underlying datasets may exhibit demographic imbalances, including disparities across race, ethnicity, age, and other population attributes, which could potentially affect model behavior and generalizability. Our method is developed and evaluated solely for research purposes and is not deployed in real-world clinical settings. While the proposed approach improves the reliability and coherence of in-context reasoning under limited information, any future clinical application would require extensive validation, careful safety evaluation, and strict compliance with medical regulations and ethical guidelines. We view this work as a step toward more responsible and transparent medical AI systems, rather than a substitute for human clinical expertise.

References

Katherine E Brown, Chao Yan, Zhuohang Li, Xinmeng Zhang, Benjamin X Collins, You Chen, Ellen Wright Clayton, Murat Kantarcioglu, Yevgeniy Vorobeychik, and Bradley A Malin. 2024. Not the models you are looking for: Traditional ml outperforms llms in clinical prediction tasks. *medRxiv*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Canyu Chen, Jian Yu, Shan Chen, Che Liu, Zhongwei Wan, Danielle Bitterman, Fei Wang, and Kai Shu. 2024. Clinicalbench: Can llms beat traditional ml models in clinical prediction? *arXiv preprint arXiv:2411.06469*.

Hongxin Ding, Yue Fang, Runchuan Zhu, Xinke Jiang, Jinyang Zhang, Yongxin Xu, Weibin Liao, Xu Chu, Junfeng Zhao, and Yasha Wang. 2025a. 3ds: Medical domain adaptation of llms via decomposed difficulty-based data selection. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 19473–19495.

Hongxin Ding, Baixiang Huang, Yue Fang, Weibin Liao, Xinke Jiang, Zheng Li, Junfeng Zhao, and Yasha Wang. 2025b. Promed: Shapley information gain guided reinforcement learning for proactive medical llms. *arXiv preprint arXiv:2508.13514*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.

Yue Fang, Yuxin Guo, Jiaran Gao, Hongxin Ding, Xinke Jiang, Weibin Liao, Yongxin Xu, Yinghao Zhu, Zhibang Yang, Liantao Ma, and 1 others. 2025. Toward better ehr reasoning in llms: Reinforcement learning with expert attention guidance. *arXiv preprint arXiv:2508.13579*.

Yue Fang, Yongxin Xu, Hongxin Ding, Junfeng Zhao, Yasha Wang, and Hongyan Jin. 2023. A method and practice for menopausal disease prediction based on knowledge graph. In *2023 IEEE International Conference on Medical Artificial Intelligence (MedAI)*, pages 10–18. IEEE.

Junyi Gao, Yinghao Zhu, Wenqing Wang, Zixiang Wang, Guiying Dong, Wen Tang, Hao Wang, Yasha Wang, Ewen M Harrison, and Liantao Ma. 2024. A comprehensive benchmark for covid-19 predictive modeling using electronic health records in intensive care. *Patterns*, 5(4).

Hila Gonen, Srini Iyer, Terra Blevins, Noah A Smith, and Luke Zettlemoyer. 2023. Demystifying prompts in language models via perplexity estimation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10136–10148.

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

SU Hongjin, Jungo Kasai, Chen Henry Wu, Weijia Shi, Tianlu Wang, Jiayi Xin, Rui Zhang, Mari Ostendorf, Luke Zettlemoyer, Noah A Smith, and 1 others. 2022. Selective annotation makes language models better few-shot learners. In *The Eleventh International Conference on Learning Representations*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2021. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421.

Alistair EW Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, and 1 others. 2023. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1.

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. 2016. Mimic-iii, a freely accessible critical care database. *Sci. Data*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Chuyuan Li, Raymond Li, Thalia S Field, and Giuseppe Carenini. 2025. Delta-knn: Improving demonstration selection in in-context learning for alzheimer's disease detection. *arXiv preprint arXiv:2506.03476*.

Weibin Liao, Yinghao Zhu, Zhongji Zhang, Yuhang Wang, Zixiang Wang, Xu Chu, Yasha Wang, and Liantao Ma. 2025. Learnable prompt as pseudo-imputation: Rethinking the necessity of traditional ehr data imputation in downstream clinical prediction. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, page 765–776.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, William B Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for gpt-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd workshop on knowledge extraction and integration for deep learning architectures*, pages 100–114.

Jiayu Liu, Zhenya Huang, Chaokun Wang, Xunpeng Huang, ChengXiang Zhai, and Enhong Chen. What makes in-context learning effective for mathematical reasoning. In *Forty-second International Conference on Machine Learning*.

Liantao Ma, Junyi Gao, Yasha Wang, Chaohe Zhang, Jiangtao Wang, Wenjie Ruan, Wen Tang, Xin Gao, and Xinyu Ma. 2020a. Adacare: Explainable clinical health status representation learning via scale-adaptive feature extraction and recalibration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 825–832.

Liantao Ma, Chaohe Zhang, Junyi Gao, Xianfeng Jiao, Zhihao Yu, Yinghao Zhu, Tianlong Wang, Xinyu Ma, Yasha Wang, Wen Tang, and 1 others. 2023. Mortality prediction with adaptive feature importance recalibration for peritoneal dialysis patients. *Patterns*, 4(12).

Liantao Ma, Chaohe Zhang, Yasha Wang, Wenjie Ruan, Jiangtao Wang, Wen Tang, Xinyu Ma, Xin Gao, and Junyi Gao. 2020b. Concare: Personalized clinical feature embedding via capturing the healthcare context. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 833–840.

Michel Minoux. 2005. Accelerated greedy algorithms for maximizing submodular set functions. In *Optimization Techniques: Proceedings of the 8th IFIP Conference on Optimization Techniques Würzburg, September 5–9, 1977*, pages 234–243. Springer.

Tai Nguyen and Eric Wong. 2023. In-context example selection with influences. *arXiv preprint arXiv:2302.11042*.

Keqin Peng, Liang Ding, Yancheng Yuan, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. 2024. Revisiting demonstration selection strategies in in-context learning. In *ACL (1)*.

Chengwei Qin, Aston Zhang, Chen Chen, Anirudh Dagar, and Wenming Ye. 2024. In-context learning with iterative demonstration selection. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7441–7455.

Stephen Robertson, Hugo Zaragoza, and 1 others. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Olle Ten Cate, Eugène JFM Custers, and Steven J Durning. 2017. Principles and practice of case-based clinical reasoning education: a method for preclinical students.

Vincent A Traag, Ludo Waltman, and Nees Jan Van Eck. 2019. From louvain to leiden: guaranteeing well-connected communities. *Scientific reports*, 9(1):1–12.

Xidong Wang, Guiming Chen, Song Dingjie, Zhang Zhiyi, Zhihong Chen, Qingying Xiao, Junying Chen, Feng Jiang, Jianquan Li, Xiang Wan, and 1 others. 2024. Cmb: A comprehensive medical benchmark in chinese. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6184–6205.

Brian J Wells, Kevin M Chagin, Amy S Nowacki, and Michael W Kattan. 2013. Strategies for handling missing data in electronic health record derived data. *Egems*, 1(3):1035.

Yongxin Xu, Xu Chu, Kai Yang, Zhiyuan Wang, Peinie Zou, Hongxin Ding, Junfeng Zhao, Yasha Wang, and Bing Xie. 2023a. Seqcare: Sequential training with external medical knowledge graph for diagnosis prediction in healthcare data. In *Proceedings of the ACM Web Conference 2023*, pages 2819–2830.

Yongxin Xu, Xinke Jiang, Xu Chu, Yuzhen Xiao, Chaohe Zhang, Hongxin Ding, Junfeng Zhao, Yasha Wang, and Bing Xie. 2024. Protomix: Augmenting health status representation learning via prototype-based mixup. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3633–3644.

Yongxin Xu, Kai Yang, Chaohe Zhang, Peinie Zou, Zhiyuan Wang, Hongxin Ding, Junfeng Zhao, Yasha Wang, and Bing Xie. 2023b. Vecocare: Visit sequences-clinical notes joint learning for diagnosis prediction in healthcare data. In *IJCAI*, volume 23, pages 4921–4929.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Zhihao Yu, Chu Xu, Yujie Jin, Yasha Wang, and Junfeng Zhao. 2024. Smart: Towards pre-trained missing-aware model for patient health status prediction. *Advances in Neural Information Processing Systems*, 37:63986–64009.

Jinyang Zhang, Yue Fang, Hongxin Ding, Weibin Liao, Muyang Ye, Xu Chu, Junfeng Zhao, and Yasha Wang. 2025a. Adept: Continual pretraining via adaptive expansion and dynamic decoupled tuning. *arXiv preprint arXiv:2510.10071*.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, and 1 others. 2025b. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*.

Yinghao Zhu, Junyi Gao, Zixiang Wang, Weibin Liao, Xiaochen Zheng, Lifang Liang, Miguel O Bernabeu, Yasha Wang, Lequan Yu, Chengwei Pan, and 1 others. 2024a. Clinicrealm: Re-evaluating large language models with conventional machine learning for non-generative clinical prediction tasks. *arXiv preprint arXiv:2407.18525*.

Yinghao Zhu, Wenqing Wang, Junyi Gao, and Liantao Ma. 2024b. Pyehr: A predictive modeling toolkit for electronic health records.

Appendix Contents

A Related Work

LLMs for Clinical Reasoning on EHRs. Recent years have witnessed growing interest in applying LLMs to clinical reasoning over EHRs. Early efforts primarily focused on adapting LLMs to the medical domain through continual pre-training or post-training on clinical corpora (Zhang et al., 2025a; Ding et al., 2025a; Fang et al., 2025; Ding et al., 2025b). However, several empirical studies have shown that, even after domain adaptation, LLMs remain limited in modeling longitudinal EHR data and activating clinically meaningful temporal patterns stored in parametric memory (Brown et al., 2024; Chen et al., 2024; Zhu et al., 2024a). These limitations motivate the exploration of inference-time adaptation strategies.

Demonstration Selection for ICL. In-context learning (ICL) (Brown et al., 2020) enables LLMs to adapt at inference time by conditioning on a small set of demonstrations, and has been widely studied in both general NLP and domain-specific settings. Existing demonstration selection methods can be broadly grouped into two paradigms. Data-perspective approaches retrieve demonstrations based on embedding similarity, assuming that representation-level proximity correlates with reasoning utility (Liu et al., 2022; Hongjin et al., 2022; Robertson et al., 2009; Yu et al., 2024). Model-perspective approaches instead leverage LLM-internal signals, such as perplexity, conditional entropy, gradients, or influence estimates, to assess the usefulness of individual demonstrations (Gonen et al., 2023; Peng et al., 2024; Liu et al.; Li et al., 2025; Qin et al., 2024). However, existing demonstration selection methods largely lack a principled framework that jointly incorporates clinical knowledge and model-aware reasoning signals for EHR-based clinical reasoning.

B Datasets

In this section, we provide a detailed introduction of the datasets used in our study, along with the preprocessing procedures applied to the structured EHR data.

Dataset Description Our experiments are conducted on two real-world EHR benchmarks: **MIMIC-III** (Johnson et al., 2016) and **MIMIC-IV** (Johnson et al., 2023). We use MIMIC-III v1.4 and MIMIC-IV v2.2, which are large-scale, publicly available critical-care databases collected at the Beth Israel Deaconess Medical Center. Both datasets provide longitudinal, de-identified ICU records with heterogeneous clinical events (e.g., vital signs, laboratory tests, and medication administrations), which naturally form irregular and long-horizon patient trajectories. This setting poses substantial challenges for clinical reasoning and makes MIMIC-III and MIMIC-IV particularly suitable benchmarks for evaluating EHR-oriented ICL methods.

Data Preprocessing We preprocess both datasets using a standardized open-source pipeline, following common practices in recent EHR benchmarks (Gao et al., 2024; Zhu et al., 2024b). For both *MIMIC-III* and *MIMIC-IV*, time-stamped clinical events within each ICU stay are temporally aggregated into daily records. For long ICU stays, we retain the most recent seven days to capture clinically salient trajectories, while summarizing earlier records to control sequence length. Missing values are handled using the Last Observation Carried Forward (LOCF) strategy (Wells et al., 2013) to maintain temporal coherence. All patient records are then chronologically ordered and organized into visit-level sequences for downstream reasoning tasks. Due to computational resource constraints, we randomly sample 2,000 patients from each dataset and partition them into training, validation, and test sets with a ratio of 8:1:1 using stratified sampling. To ensure fair and consistent evaluation, the test set is fixed and shared across all experiments. Dataset statistics are reported in Table 5.

C Tasks

Definition 1 (EHR Dataset). A patient's EHR is represented as a sequence of T time-ordered visits $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_T]$, where each visit $\mathbf{x}_t = \{l_{t,1}, l_{t,2}, \cdots, l_{t,n_t}\}$ contains n_t lab test features.

Definition 2 (Mortality Prediction). Given $\mathbf{X}$, predict whether the patient will survive the hospital stay. The label $y \in \{0, 1\}$ denotes death ($y = 1$) or survival ($y = 0$).

Definition 3 (Readmission Prediction). Given $\mathbf{X}$, predict whether the patient will be readmitted within 30 days after discharge. The label $y \in \{0, 1\}$ indicates readmission ($y = 1$) or not ($y = 0$).

Definition 4 (Length-of-Stay Prediction). Given $\mathbf{X}$, predict the length of the patient's hospital stay. Following prior work (Harutyunyan et al., 2019), we formulate this task as a classification problem, where the label $y \in \{1, 2, \cdots, C\}$ indicates the length-of-stay category corresponding to predefined time intervals.

D Baseline Implementation Details

We compare *GraphWalker* with representative baselines spanning multiple paradigms. In this subsection, we provide implementation details for each baseline. For all methods, we strictly constrain the LLM to directly output the final prediction in the task-specific format, without generating any intermediate reasoning or explanations, following the prompt templates described in Appendix J.

- **Vanilla ICL baselines.** *Zero-shot* corresponds to a special case of ICL where no demonstration examples are provided, and the LLM directly produces predictions based on the task description and patient EHR data. *Random* samples a fixed number of patient examples uniformly at random as demonstrations. Both baselines adopt the same prompt structure as described in Appendix J, differing only in whether in-context examples are included.

Dataset	MIMIC-III				MIMIC-IV				
	Total	Alive	Dead	Avg. LOS	Total	Alive	Dead	Re.	No Re.
Test Set Statistics									
# Patients	200	165	35	3.8	200	182	18	53	147
# Total visits	8791	7167	1624	-	679	610	69	170	507
# Avg. visits	44.0	43.4	46.4	-	3.4	3.4	3.8	3.2	3.4
Training Set Statistics									
# Patients	1600	1405	195	3.7	1600	1465	135	391	1209
# Total visits	70641	61713	8928	-	5848	5346	502	1430	4432
# Avg. visits	44.2	43.9	45.8	-	3.7	3.6	3.7	3.7	3.7
Validation Set Statistics									
# Patients	200	171	29	3.7	200	181	19	51	149
# Total visits	8918	7669	1249	-	712	652	60	178	537
# Avg. visits	44.6	44.8	43.1	-	3.6	3.6	3.2	3.5	3.6

Table 5: *Statistics of the MIMIC-III dataset and MIMIC-IV dataset.* "Re." stands for Readmission, indicating patients who are readmitted to the ICU within 30 days of discharge, while "No Re." represents patients who are not readmitted.

- **Supervised.** *SFT* adapts LLMs to downstream tasks by fine-tuning model parameters on task-specific data. Specifically, we organize the training set into question–answer pairs using the same prompt templates described in Appendix J, and apply parameter-efficient fine-tuning via LoRA (Hu et al., 2022) to update the backbone LLM.
- **Data Perspective Selection.** *Semantic-emb* retrieves top-k demonstrations based on similarity in the embedding space of *Qwen3-Embedding-8B* (Zhang et al., 2025b), a widely used and strong general-purpose semantic model. For efficiency, we compute embeddings using vLLM (Kwon et al., 2023) to accelerate large-scale embedding inference. *EHR-emb* retrieves top-k demonstrations based on similarity computed from embeddings generated by *SMART* (Yu et al., 2024), a pretrained model specifically designed for EHR data. We follow the hyperparameter settings recommended in the original work and use the `[CLS]` token embedding as the patient representation, as it encodes global semantic information of the longitudinal EHR sequence.
- **Model Perspective Selection.** *SPELL* (Gonen et al., 2023) selects demonstrations by ranking candidates with LM perplexity (i.e., teacher-forced next-token cross-entropy) and choosing the lowest-perplexity top-k examples. In our implementation, we score all training examples with the same causal LM and select the top-k demonstrations by sorting the average loss in ascending order. The resulting few-shot set is shared across all test instances and inserted into the ICL prompt as demonstrations, without per-test-instance re-selection.

  Influence (Nguyen and Wong, 2023) estimates the importance of demonstrations by measuring their marginal contribution to the model's performance on a validation set. It calculates influence scores based on the average performance difference between sampled subsets that include a specific example and those that exclude it.

  IDS (Qin et al., 2024) selects demonstrations by calculating the semantic similarity between training examples and reasoning paths generated through Zero-shot CoT. The framework iteratively refines the demonstration set by utilizing the model's self-generated reasoning from the previous round as an updated query for subsequent retrieval.

  CONE (Peng et al., 2024) selects demonstrations by minimizing the conditional entropy of the test input on a per-example basis, without modeling complementarity among selected examples. Following the original implementation, we first perform a semantic pre-filtering step using *Qwen3-Embedding-8B* to retrieve the top-10 semantically similar candidates. We then select the top-k demonstrations with the largest reduction in conditional entropy, where the entropy is computed independently for each candidate example.

  LMS3 (Liu et al.) selects demonstrations by balancing LLM-oriented semantic similarity and inference stability, derived from a theoretical bound on reasoning efficacy. It uniquely incorporates a demonstration rejection mechanism that adaptively filters out unsuitable examples to prevent negative transfer.

  Delta-KNN (Li et al., 2025) selects demonstrations by prioritizing candidates that were empirically observed to be most helpful for similar inputs, using a cached example-to-example gain matrix. In our implementation, we first take a fixed training subset of examples to construct the Delta matrix via paired zero-shot vs. one-shot inference, and cache the resulting matrix for each dataset/LLM setting. At test time, we use *Qwen3-Embedding-8B* to retrieve the top-10 semantically similar candidates, then re-rank them with the Delta-based scores and select the final top-k demonstrations. The selected three demonstrations are re-selected per test instance and inserted into the ICL prompt as demonstrations for final inference.

E Experimental Setup of Pilot Study

To investigate the behavior of different ICL strategies for EHR reasoning, we adopt *Qwen3-14B* (Yang et al., 2025) as the backbone LLM for all pilot experiments. We conduct experiments on two widely used, large-scale public EHR benchmarks: *MIMIC-III* (Johnson et al., 2016) and *MIMIC-IV* (Johnson et al., 2023), and focus on two representative downstream tasks: *mortality prediction* and *readmission prediction*. For each task, the LLM is prompted in a few-shot setting using demonstrations selected by different ICL strategies. Detailed dataset statistics, preprocessing procedures, task definitions, and prompt templates are provided in Appendix B and J.

Definition of Information Gain ΔH. To quantitatively analyze how different demonstration selection strategies affect the model's understanding of a test query, we measure information gain using an entropy-based criterion. Unlike the marginal gain formulation used in our method, the ΔH reported in the pilot study is defined *relative to the zero-shot baseline*.

Formally, let x_{test} denote an unlabeled test input, and let $\mathcal{S}_k$ denote a demonstration set of size k selected by a given ICL strategy. We define the conditional entropy of the model's prediction on x_{test} under a demonstration composition $\mathcal{S}_k$ as

$$H_\theta(x_{\text{test}} \mid \mathcal{S}_k) = H_\theta(x_{\text{test}}, \mathcal{S}_k) - H_\theta(\mathcal{S}_k), \quad (13)$$

where $H_\theta(x_{\text{test}}, \mathcal{S}_k)$ denotes the cross-entropy of the full prompt (including both the demonstrations and the test query), and $H_\theta(\mathcal{S}_k)$ denotes the cross-entropy of the demonstration composition alone. This conditional entropy reflects the model's predictive uncertainty and thus its degree of understanding of the test query under the given context.

We then define the information gain of a demonstration composition $\mathcal{S}_k$ relative to the zero-shot setting as

$$\Delta H_\theta(\mathcal{S}_k) = H_\theta(x_{\text{test}} \mid \emptyset) - H_\theta(x_{\text{test}} \mid \mathcal{S}_k), \quad (14)$$

where $\emptyset$ denotes the empty demonstration set (i.e., zero-shot prompting). A larger $\Delta H_\theta(\mathcal{S}_k)$ indicates that the selected demonstrations reduce the model's uncertainty more effectively compared to the zero-shot baseline.

For each ICL strategy and each $k \in \{1, 2, 3, 4\}$, we compute $\Delta H_\theta(\mathcal{S}_k)$ for every validation instance and report the average information gain over the validation set.

F Implementation Details

For patient graph construction, we set the number of nearest neighbors to $k_g = 8$. For instance-aware cohort retrieval, the number of retrieved cohorts is set to $K_c = 3$, and within each retrieved cohort, we select $K_a = 3$ anchor nodes to initialize the search frontier. A detailed analysis and discussion of these key hyperparameters are provided in Appendix M. For graph-based cohort discovery, the resolution parameter of the Leiden algorithm is set to 0.9. All experiments are repeated three times with different random seeds, and we report the mean performance along with the standard deviation. To accelerate the computation of conditional entropy during LLM-guided demonstration selection, we adopt `vLLM` (Kwon et al., 2023) for efficient inference.

G Computational Resources and Software Environment

All experiments were conducted on a server equipped with two NVIDIA H20 GPUs (96 GB memory each) and 503 GB system RAM, running Ubuntu 22.04.5 LTS. The software environment was based on Python 3.11.11 with Conda 23.5.2, and experiments were implemented using PyTorch 2.6.0, HuggingFace Transformers 4.51.3, and SpaCy 3.8.4, all with default settings unless otherwise specified.

H Algorithm

Please see Algorithm 1. For clarity, Algorithm 1 presents the full greedy implementation of *GraphWalker*. In practice, we adopt a lazy greedy strategy to improve computational efficiency, which is discussed in detail in Appendix I.

I Lazy Greedy Search

Motivation and Complexity. Algorithm 1 presents a full greedy search, which recomputes the marginal information gain for *every* candidate in the frontier at each iteration. Let $F_t^{(q)}$ denote the frontier size at step t and K be the ICL budget. Full greedy therefore requires $\sum_{t=1}^{K} |F_t^{(q)}|$ exact evaluations of $\Delta H(\cdot \mid \mathcal{S}_{t-1}^{(q)})$, resulting in $O(K\bar{F})$ LLM forward passes, where $\bar{F}$ denotes the average frontier size.

To reduce this cost, we adopt the standard *lazy greedy* acceleration (Minoux, 2005), which maintains cached upper bounds of marginal gains in a priority queue and re-evaluates a candidate only when it reaches the top. Unlike full greedy, lazy greedy avoids recomputing marginal gains for all frontier nodes at every step. In practice, this reduces the number of exact marginal evaluations from $O(\sum_t |F_t^{(q)}|)$ to approximately $O(|F_0^{(q)}| + R)$, where R is the number of necessary re-evaluations during the search and is typically much smaller than $\sum_t |F_t^{(q)}|$ when marginal gains exhibit diminishing returns. The additional overhead introduced by the priority queue is $O(\log |F_t^{(q)}|)$ per update, which is negligible compared to LLM inference.

Assumption (Monotone Submodularity on the Frontier). Lazy greedy is exact when the set objective is *monotone submodular*. In our setting, we define the objective as the *prompt-level information gain* on the test instance q:

$$f_q(\mathcal{S}) \triangleq \Delta H(\mathcal{S}; q) = \ell\Big(\mathcal{S}^{(q)} = \emptyset\Big) - \ell(\mathcal{S}), \quad (15)$$

where $\ell(\mathcal{S})$ is the token-level cross-entropy of the test case conditioned on demonstrations $\mathcal{S}$. The marginal gain is

$$\Delta_q(v \mid \mathcal{S}) \triangleq f_q(\mathcal{S} \cup \{v\}) - f_q(\mathcal{S}) = \ell(\mathcal{S}) - \ell(\mathcal{S} \cup \{v\}). \tag{16}$$

We assume $\Delta_q(v \mid \mathcal{S})$ is *approximately diminishing* as $\mathcal{S}$ grows, i.e., for $\mathcal{A} \subseteq \mathcal{B}$, $\Delta_q(v \mid \mathcal{A}) \geq \Delta_q(v \mid \mathcal{B})$. This assumption is consistent with our empirical observations: adding demonstrations often yields diminishing reductions in predictive uncertainty due to redundancy among clinically similar cases. Even when submodularity is only approximate, lazy greedy remains a widely used heuristic for accelerating greedy selection.

Implementation Details (Frontier Expansion + Priority Queue). Our implementation follows the standard lazy greedy template with a max-priority queue. At each step, we pop the candidate with the largest *cached* marginal gain (an upper bound under submodularity), recompute its *true* marginal gain w.r.t. the current set $\mathcal{S}^{(q)}$, and compare it with the next cached best. If it remains the best, we accept it; otherwise, we update its key and push it back. Upon selecting $v^\star$, we update $\mathcal{S}^{(q)} \leftarrow \mathcal{S}^{(q)} \cup \{v^\star\}$, remove $v^\star$ from the frontier, and expand the frontier by adding its graph neighbors $\mathcal{N}(v^\star)$ while excluding already selected nodes. We further apply early stopping when the best marginal gain becomes non-positive (i.e., $\Delta_q(v^\star \mid \mathcal{S}^{(q)}) \leq 0$), preventing misleading demonstrations.

J Prompt Details

In this section, we detail the prompt designs used in our framework. We implement two prompting schemes: a *zero-shot* template, which provides no in-context examples and requires the LLM to directly produce the task-specific output, and a *few-shot* template, which augments the prompt with a set of labeled patient demonstrations to support ICL. Both templates share a unified structure for presenting longitudinal EHR data. In addition, we specify task-specific instructions for each prediction task, including in-hospital mortality, readmission, and length-of-stay prediction, ensuring consistent and fair evaluation across different prompting settings.

Prompt for Mortality Prediction

You are tasked with predicting in-hospital mortality based on patient EHR data.
Provide only a floating-point number between 0 and 1 representing the predicted probability of mortality (a higher value indicates a higher likelihood of death).
Do **not** provide any reasoning, explanation, or additional text. Output **only** the numerical value.
Example: `0.XX`

Prompt for Readmission Prediction

You are tasked with predicting whether a patient will be readmitted within 30 days after hospital discharge based on EHR data.
Provide only a floating-point number between 0 and 1 representing the predicted probability of 30-day readmission after discharge (including cases where the patient dies within 30 days, which are counted as readmission events).
Do **not** provide any reasoning, explanation, or additional text. Output **only** the numerical value.
Example: `0.XX`

Prompt for Length-of-Stay (LOS) Prediction

You are tasked with predicting the patient's length of hospital stay based on EHR data.
Provide only a single letter (**A**, **B**, **C**, or **D**) representing the predicted length-of-stay category:

- **A**: Less than 3 days (< 3 days)
- **B**: 3 to 7 days (3–7 days)
- **C**: 7 to 14 days (7–14 days)
- **D**: More than 14 days (> 14 days)

Do **not** provide any reasoning, explanation, or additional text. Output **only** the letter (**A**, **B**, **C**, or **D**).
Example: `B`

Zero-shot Prompt Template

You will be provided with longitudinal electronic health record (EHR) data for a single patient. Each clinical feature is represented as a time-ordered list of measurements corresponding to the same hospital stay. Missing values are denoted as `NaN`. Units and reference ranges are provided where applicable.
Patient Information:

- Number of measurements: `{LENGTH}`
- Measurement times (hours since admission): `[{RECORD_TIME_LIST}]`

Task Description:
`{TASK_DESCRIPTION}`
Clinical Features Over Time:

- **Heart Rate** (Unit: bpm. Reference range: 60–100):
  `[82, 86, 90, 95, NaN, 88, 84, . . . ]`
- **Systolic Blood Pressure** (Unit: mmHg. Reference range: < 120):
  `[110, 118, 125, 132, 120, NaN, 115, . . . ]`
- **Oxygen Saturation** (Unit: %. Reference range: 95–100):
  `[100, 99, 98, NaN, 96, 95, 97, . . . ]`
- **Glasgow Coma Scale (Total)** (Unit: /. Reference range: /.):
  `[15, 14, NaN, 12, 10, 8, . . . ]`
- ⋮

Your Answer:
`{MODEL_OUTPUT}`

Few-shot Prompt Template

You will be provided with longitudinal electronic health record (EHR) data for a single patient. Each clinical feature is represented as a time-ordered list of measurements corresponding to the same hospital stay. Missing values are denoted as `NaN`. Units and reference ranges are provided where applicable.
Patient Information:

- Number of measurements: `{LENGTH}`
- Measurement times (hours since admission): `[{RECORD_TIME_LIST}]`

Task Description:
`{TASK_DESCRIPTION}`
Instructions & Output Format:
`{RESPONSE_FORMAT}`
In-context Examples:
Below are example patient records and their corresponding labels. These examples are provided to guide the prediction for the target patient.

- **Example 1:**
  Clinical Features: `[. . . ]`
  Label: `{LABEL_1}`
- **Example 2:**
  Clinical Features: `[. . . ]`
  Label: `{LABEL_2}`
- ⋮

Target Patient:
Clinical Features Over Time:

- **Heart Rate** (Unit: bpm. Reference range: 60–100):
  `[82, 86, 90, 95, NaN, 88, 84, . . . ]`
- **Systolic Blood Pressure** (Unit: mmHg. Reference range: < 120):
  `[110, 118, 125, 132, 120, NaN, 115, . . . ]`
- **Oxygen Saturation** (Unit: %. Reference range: 95–100):
  `[100, 99, 98, NaN, 96, 95, 97, . . . ]`
- **Glasgow Coma Scale (Total)** (Unit: /. Reference range: /.):
  `[15, 14, NaN, 12, 10, 8, . . . ]`
- ⋮

Your Answer:
`{MODEL_OUTPUT}`

K Shot Scale Analysis

To examine the effect of shot number on the behavior of *GraphWalker*, we evaluate its performance under varying k-shot settings with $k \in \{1, 2, 3, 4\}$. Experiments are conducted on *MIMIC-III Mortality* and *MIMIC-IV Readmission* using *Qwen3-14B* as the backbone LLM, and the results are visualized in Figure 6 and 8.

We observe that *GraphWalker* consistently outperforms all baselines across different k-shot settings,

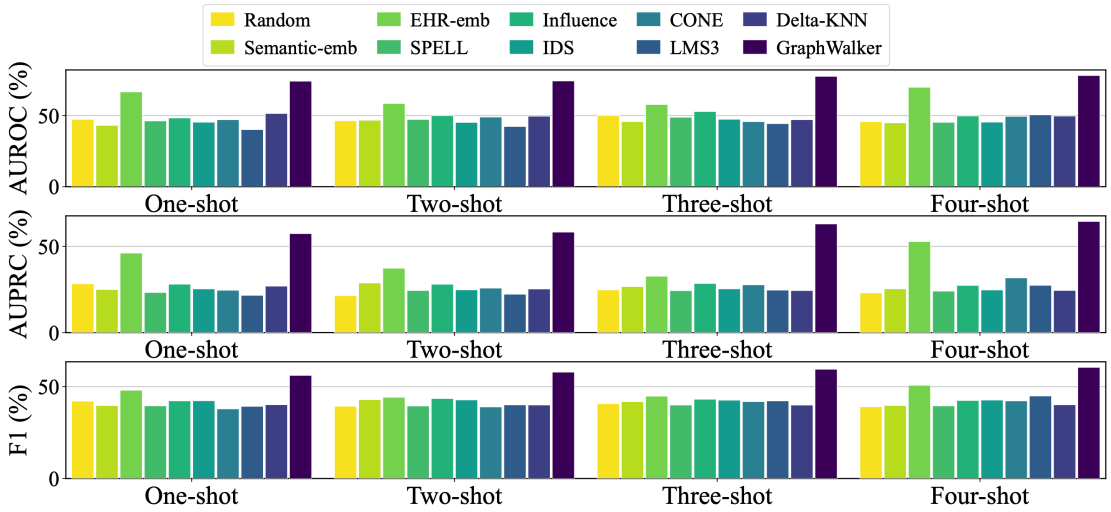

Figure 8: Shot scaling performance on MIMIC-IV Readmission with Qwen3-14B as the backbone LLM.

demonstrating strong robustness to the choice of shot number. Meanwhile, the performance of existing methods exhibits non-monotonic and dataset-dependent trends as k increases, indicating that **simply adding more demonstrations does not necessarily improve EHR reasoning performance**. In contrast, indiscriminately adding demonstrations may introduce irrelevant or misleading context, which can hinder coherent reasoning. We attribute this behavior to the inclusion of inappropriate demonstrations, which may introduce irrelevant or misleading context and thereby disrupt coherent reasoning rather than enhancing it.

Notably, ***GraphWalker* is the only approach whose performance improves consistently with increasing k.** This indicates that, by grounding demonstration selection in graph-based discovery of clinically similar patient groups and performing progressive, LLM-guided search over a reliable candidate set, *GraphWalker* effectively identifies in-context example combinations that maximize information gain for the target LLM.

Another notable observation is that purely model-perspective methods (e.g., CONE, LMS3, SPELL) consistently struggle to achieve competitive performance compared to *EHR-emb* and *GraphWalker*. This suggests that **relying solely on LLM-internal signals is insufficient to construct reliable and informative candidate sets for EHR reasoning**, as such signals lack grounding in clinically meaningful patient distributions.

L Applicability to More Tasks

To further examine the generality of *GraphWalker* beyond EHR-centric prediction tasks, we extend our evaluation to two additional medical reasoning benchmarks: **MedQA** (Jin et al., 2021) and **CMB** (Wang et al., 2024).

MedQA consists of medical examination questions designed to assess diagnostic reasoning and clinical decision-making, with each question grounded in a realistic patient case description. Although the full benchmark is multilingual, we focus on the English subset, which contains approximately 12.7K case-based questions and has been widely adopted for evaluating LLM-based medical reasoning.

CMB is a large-scale Chinese medical benchmark covering a wide range of clinical specialties and question types. Following (Ding et al., 2025b), we filter the dataset to retain only case-based questions involving explicit patient scenarios, ensuring suitability for in-context reasoning and demonstration-based evaluation. Dataset statistics are summarized in Table 6.

In this experimental setting, both *GraphWalker* and the Semantic-embedding baseline employ *Qwen3-Embedding-8B* as the retrieval encoder, while *Qwen3-14B* is used as the backbone LLM for all methods. As shown in Table 4, *GraphWalker* consistently achieves the best performance on both MedQA and CMB, indicating that its effectiveness is not limited to structured EHR prediction tasks, but also extends to medical reasoning over unstructured clinical text, **highlighting its potential for broader real-world clinical applications**.

Split	MedQA	CMB
Train	10,178	15,465
Val	1,272	1,940
Test	1,273	1,935
Total	12,723	19,340

Table 6: Dataset statistics for MedQA and CMB.

M Detailed Hyper-Parameter Analysis

We conduct a detailed analysis of several key hyper-parameters introduced in *GraphWalker* to assess its robustness and provide practical guidance for future applications. Specifically, we examine the impact of the graph neighborhood size k_g, the number of retrieved cohorts K_c, and the number of anchor nodes per cohort K_a. Unless otherwise specified, all results are reported on the validation set of *MIMIC-IV Readmission* with *Qwen3-14B* as the backbone LLM, while keeping all other settings fixed.

M.1 Graph Neighborhood Size k_g

We vary the graph neighborhood size k_g from 4 to 10, as shown in Figure 7. The results exhibit a clear non-monotonic trend. Specifically, increasing k_g from 4 to 8 consistently improves performance, indicating that moderately enlarging the local neighborhood helps capture richer population-level structure and clinically meaningful similarity relations among patients. However, further increasing k_g to 10 leads to a noticeable performance degradation. We attribute this decline to the introduction of weakly related or clinically heterogeneous neighbors when the graph becomes overly dense. Such neighbors may blur the local structure of the patient graph and propagate less informative or misleading signals during frontier expansion, thereby reducing the quality of candidate demonstrations.

M.2 Number of Retrieved Cohorts K_c

We analyze the effect of the number of retrieved cohorts K_c by varying it from 1 to 4, as shown in Figure 7. When K_c increases from 1 to 3, both AUPRC and F1 steadily improve. This indicates that retrieving multiple cohorts enables *GraphWalker* to incorporate a more diverse yet clinically relevant set of patient subpopulations, thereby enriching the candidate pool for demonstration selection. In particular, aggregating information across a small number of related cohorts helps mitigate over-specialization to a single cohort and improves robustness to intra-cohort variability. However, further increasing K_c to 4 leads to a notable performance degradation. We attribute this decline to the inclusion of loosely related or heterogeneous cohorts, which may dilute clinically meaningful signals and introduce conflicting patterns into the candidate set.

M.3 Anchor Nodes per Cohort K_a

We investigate the effect of the number of anchor nodes per cohort K_a, as shown in Figure 7. The results show that performance peaks at $K_a = 3$. This suggests that selecting a moderate number of anchors per cohort provides a good balance between intra-cohort coverage and candidate quality. With $K_a = 3$, the anchors are sufficient to represent diverse yet clinically consistent patient profiles within each cohort, enabling the subsequent frontier expansion to explore informative neighborhoods without being overwhelmed by redundant or weakly relevant cases. In contrast, larger values of K_a tend to over-expand the initial frontier with marginal or noisy anchors, which increases redundancy and heterogeneity in the candidate set and degrades the effectiveness of interaction-aware demonstration selection. These results indicate that a carefully chosen, moderate anchor size is crucial for stable and effective frontier-based search.

N Runtime Analysis

We report the end-to-end runtime of different ICL demonstration selection methods on the *MIMIC-IV Readmission* task under the 3-shot setting. The reported time includes both offline preparation (e.g., score estimation or candidate construction) and inference-time demonstration selection. All methods are evaluated using *Qwen3-14B* as the backbone LLM. We focus on representative ICL methods that require multiple LLM forward passes during selection.

Method	Total Runtime
Influence	37 min 41 s
IDS	40 min 58 s
CONE	34 min 23 s
Delta-KNN	44 min 36 s
GraphWalker	46 min 33 s

Table 7: End-to-end runtime comparison of ICL demonstration selection methods on the *MIMIC-IV Readmission* task (3-shot). All results are obtained using *Qwen3-14B*.

Discussion. As shown in Table 7, *GraphWalker* exhibits a runtime comparable to other ICL methods that require multiple LLM forward passes for demonstration evaluation. Although *GraphWalker* involves interaction-aware selection with frontier expansion, its overall time cost remains in the same order of magnitude as Influence, IDS, CONE, and Delta-KNN. This suggests that the additional modeling of demonstration interactions does not introduce prohibitive overhead, making *GraphWalker* practically applicable in real-world inference settings.

O Case Study

To qualitatively illustrate why *GraphWalker* yields more reliable in-context demonstrations for EHR prediction, we present in Table 8 a representative case from the *MIMIC-III Mortality* test set using *Qwen3-14B* as the backbone LLM, where the *Zero-shot* baseline produces a severely miscalibrated prediction, while *GraphWalker* corrects it.

Algorithm 1 *GraphWalker*

Require: Training EHR dataset $\{\mathbf{X}_i\}_{i=1}^{N}$; test patient record $\mathbf{X}^{(q)}$; pretrained EHR encoder $\mathcal{M}_{\text{exp}}$; LLM M_θ; graph neighbor size k_g; number of retrieved cohorts K_c; anchors per cohort K_a; demonstration budget K.

# Stage 1: Patient Graph Construction and Cohort Discovery
for $i \leftarrow 1$ to N **do**
    $\mathbf{h}_i \leftarrow \mathcal{M}_{\text{exp}}(\mathbf{X}_i)$
end for
Construct a (symmetrized) k_g-NN patient graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ over $\{\mathbf{h}_i\}$ using cosine similarity
$\{\mathcal{C}_m\}_{m=1}^{M} \leftarrow \textsc{Leiden}(\mathcal{G})$
for $m \leftarrow 1$ to M **do**
    $\mathbf{z}_m \leftarrow \frac{1}{|\mathcal{C}_m|} \sum_{i \in \mathcal{C}_m} \mathbf{h}_i$
end for
# Stage 2: Instance-aware Cohort Retrieval and Anchor Initialization
$\mathbf{h}^{(q)} \leftarrow \mathcal{M}_{\text{exp}}(\mathbf{X}^{(q)})$
$\mathcal{C}_{\text{ret}}^{(q)} \leftarrow \text{Top-}K_c\big(\text{sim}(\mathbf{h}^{(q)}, \mathbf{z}_m)\big)$
$\mathcal{F}^{(q)} \leftarrow \emptyset$
for each cohort $\mathcal{C}_m \in \mathcal{C}_{\text{ret}}^{(q)}$ **do**
    $\mathcal{V}_m^{(q)} \leftarrow \text{Top-}K_a\big(\text{sim}(\mathbf{h}^{(q)}, \mathbf{h}_i)\big),\ v_i \in \mathcal{C}_m$
    $\mathcal{F}^{(q)} \leftarrow \mathcal{F}^{(q)} \cup \mathcal{V}_m^{(q)}$
end for
# Stage 3: LLM-guided Greedy Search with Frontier Expansion
$\mathcal{S}^{(q)} \leftarrow \emptyset$
$H_0 \leftarrow H_\theta(\mathbf{X}^{(q)} \mid \mathcal{S}^{(q)})$
while $|\mathcal{S}^{(q)}| < K$ **and** $\mathcal{F}^{(q)} \neq \emptyset$ **do**
    $v^* \leftarrow \text{None}$; $\Delta^* \leftarrow 0$; $H^* \leftarrow H_0$
    for each $v_i \in \mathcal{F}^{(q)}$ **do**
        $H_i \leftarrow H_\theta(\mathbf{X}^{(q)} \mid \mathcal{S}^{(q)} \cup \{v_i\})$
        $\Delta H_\theta(v_i \mid \mathcal{S}^{(q)}) \leftarrow H_0 - H_i$
        if $\Delta H_\theta(v_i \mid \mathcal{S}^{(q)}) > \Delta^*$ **then**
            $v^* \leftarrow v_i$; $\Delta^* \leftarrow \Delta H_\theta(v_i \mid \mathcal{S}^{(q)})$; $H^* \leftarrow H_i$
        end if
    end for
    if $v^* = \text{None}$ **or** $\Delta^* \leq 0$ **then**
        break
    end if
    $\mathcal{S}^{(q)} \leftarrow \mathcal{S}^{(q)} \cup \{v^*\}$
    $H_0 \leftarrow H^*$
    $\mathcal{F}^{(q)} \leftarrow \mathcal{F}^{(q)} \setminus \{v^*\}$
    $\mathcal{F}^{(q)} \leftarrow \mathcal{F}^{(q)} \cup \mathcal{N}(v^*)$
    $\mathcal{F}^{(q)} \leftarrow \mathcal{F}^{(q)} \setminus \mathcal{S}^{(q)}$
end while
return $\mathcal{S}^{(q)}$

Table 8: Case study on *MIMIC-III Mortality*.

Component	Content
Prompt	You will be provided with longitudinal electronic health record (EHR) data for a single patient. Each clinical feature is represented as a time-ordered list of measurements corresponding to the same hospital stay. Missing values are denoted as `NaN`. **Patient Information:** • Number of measurements: `19` • Measurement times (hours since admission): `[4.68, 5.18, 6.18, 7.18, 8.18, . . . ]` **Task Description:** You are tasked with predicting in-hospital mortality based on patient EHR data. **Instructions & Output Format:** Provide only a floating-point number between 0 and 1 representing the predicted probability of mortality. Do **not** provide any reasoning or additional text. Output **only** the numerical value (e.g., `0.XX`).
Target Patient	**Clinical Features Over Time:** • **Heart Rate (bpm):** `[117.0, 110.0, 111.0, 107.0, 102.0, . . . , 119.0, 122.0, 118.0]` • **Systolic BP (mmHg):** `[99.0, 96.0, 92.0, 109.0, 87.0, . . . , 127.0, 118.0, 120.0]` • **Diastolic BP (mmHg):** `[39.0, 56.0, 44.0, 54.0, 47.0, . . . , 87.0, 75.0, 67.0]` • **Mean BP (mmHg):** `[59.0, 69.3, 60.0, 72.3, 60.3, . . . , 100.3, 89.3, 84.7]` • **SpO_2 (%):** `[97.0, 94.0, 95.0, 95.0, 94.0, . . . , 96.0, 95.0, 95.0]` • **GCS (Total):** `[NaN, 15.0, 15.0, 15.0, 15.0, . . . , 15.0, 15.0, 15.0]` ⋮
Ground Truth	`0` (Survival).
Zero-shot Output	**Prediction:** `0.72` → **Incorrect**
***GraphWalker* Examples**	**Example 1** *Label:* `0`*;* • **Heart Rate:** `[78.0, 104.0, 81.0, 98.0, 74.0, . . . , 73.0, NaN, 74.0, 72.0]` • **Systolic BP:** `[119.0, 134.0, 111.0, 165.0, 149.0, . . . , 107.0, 96.0, 112.0, 118.0]` • **Diastolic BP:** `[62.0, 70.0, 79.0, 73.0, 66.0, . . . , 43.0, 56.0, 62.0]` • **SpO_2:** `[98.0, 94.0, 95.0, 98.0, 96.0, . . . , 98.0, NaN, 97.0, 96.0]` • **GCS (Total):** `[NaN, 14.0, NaN, 14.0, NaN, . . . , 15.0, . . . ]` ⋮ **Example 2** *Label:* `0`*;* • **Heart Rate:** `[87.0, 92.0, 92.0, 97.0, 90.0, . . . , 88.0, 77.0, 75.0, 74.0]` • **Systolic BP:** `[138.0, 155.0, 143.0, 152.0, 131.0, . . . , 136.0, 116.0, 133.0, 136.0]` • **Diastolic BP:** `[62.0, 78.0, 69.0, 79.0, 57.0, . . . , 59.0, 66.0, 61.0]` • **SpO_2:** `[93.0, 93.0, 93.0, 95.0, 91.0, . . . , 93.0, 97.0, 98.0, 98.0]` • **GCS (Total):** `[NaN, 15.0, NaN, . . . , 15.0, . . . ]` ⋮ **Example 3** *Label:* `0`*;* • **Heart Rate:** `[102.0, 105.0, 95.0, 91.0, 96.0, . . . , 77.0, 79.0, 80.0, 81.0]` • **Systolic BP:** `[122.0, 118.0, 115.0, 118.0, 106.0, . . . , 132.0, 118.0, 129.0, 130.0]` • **Diastolic BP:** `[62.0, 52.0, 65.0, 74.0, 63.0, . . . , 84.0, 76.0, 79.0, 85.0]` • **SpO_2:** `[83.0, 95.0, 94.0, 95.0, 93.0, . . . , 93.0, 93.0, 92.0, 91.0]` • **GCS (Total):** `[NaN, NaN, 15.0, . . . , 15.0, . . . ]` ⋮
***GraphWalker* Output**	**Prediction:** `0.0689` → **Correct**

P Notations Table

This section summarizes the key notations used in the *GraphWalker* framework.

Symbol	Description
$\mathbf{X}_i$	EHR record of patient i
$\mathcal{M}_{\text{exp}}$	Pretrained EHR encoder
$\mathbf{h}_i$	Patient embedding of patient i
N	Number of patients in the EHR base
$\mathcal{G}$	Population-level patient graph
$\mathcal{V}$	Set of patient nodes in the graph
$\mathcal{E}$	Set of edges constructed via k_g-nearest neighbors in the embedding space
k_g	Number of nearest neighbors used for graph construction
$\text{sim}(\cdot,\cdot)$	Cosine similarity function between embeddings
$\mathcal{C}_m$	Cohort subgraph discovered from the patient graph via Leiden algorithm
M	Total number of discovered cohorts
$\mathbf{z}_m$	Centroid embedding of cohort $\mathcal{C}_m$ obtained by mean pooling
$\mathcal{Q}$	Graph modularity objective optimized during cohort discovery
$\mathbf{X}^{(q)}$	EHR record of the target patient
$\mathbf{h}^{(q)}$	Embedding of the target patient
$\mathcal{C}_{\text{ret}}^{(q)}$	Set of top-K_c cohorts retrieved for the target patient
K_c	Number of cohorts retrieved for the target patient
$\mathcal{V}_m^{(q)}$	Set of anchor nodes selected from cohort $\mathcal{C}_m$ for the target patient
K_a	Number of anchor nodes selected per retrieved cohort
$\mathcal{F}^{(q)}$	Search frontier consisting of candidate demonstration nodes for the target patient
$\mathcal{S}^{(q)}$	Current demonstration set constructed for the target patient
K	Maximum number of demonstrations
v_i	Candidate demonstration node corresponding to patient i
v^*	Selected demonstration node that maximizes marginal information gain
x_{test}	Unlabeled test query constructed from the target patient's EHR record
$H_\theta(x_{\text{test}} \mid \mathcal{S})$	Conditional entropy of the LLM's prediction on the test query given demonstration set $\mathcal{S}$
$\Delta H_\theta(v_i \mid \mathcal{S})$	Marginal information gain obtained by adding candidate v_i to demonstration set $\mathcal{S}$
θ	Parameters of the target LLM
$\mathcal{N}(v)$	Neighbor set of node v in the patient graph, used for frontier expansion

Table 9: Key Notations Used in *GraphWalker*

Q Code and Data Availability

To support reproducibility and facilitate future research, we will publicly release the full implementation of *GraphWalker* along with the processed datasets used in our experiments upon publication.

R Use of Large Language Models

In this work, large language models (LLMs) were used in a supportive role to assist with language refinement and programming-related tasks, including improving clarity, grammatical correctness, and presentation quality, as well as providing high-level guidance for code structuring and debugging. All LLM-assisted outputs were carefully reviewed, verified, and, when necessary, revised by the authors prior to inclusion. The core research ideas, methodological design, experimental setup, and result analysis were conceived and carried out entirely by the authors. LLMs did not contribute to the formulation of scientific hypotheses, the design of the proposed methods, or the derivation of research conclusions.