SEQUENCE COMPOSITION IN EHR FOUNDATION MODEL PRETRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Autoregressive electronic health record (EHR) foundation models predict the next token in sequences of structured medical events. A common pretraining strategy concatenates patient histories into a global token stream and samples fixed-length windows from it. This gives patients with longer histories more training exposure and, under standard causal attention, allows tokens to attend across patient boundaries. These choices can misalign pretraining with downstream clinical prediction, where each prediction is based on a single patient's history. We therefore investigate how the sampling and assembly of patient histories into pretraining sequences affect clinical prediction performance. We use the conventional global-stream approach as a baseline and compare alternative sequence composition methods that control how patient histories contribute to pretraining sequences. Across four MIMIC-IV v2.2 tasks, our best method improves macro AUROC, AUPRC, F1, and balanced accuracy, including relative gains of 9.3% in F1 and 2.7% in AUPRC under a fixed training recipe. We show that these gains extend across four additional dataset and tokenization settings, including INSPIRE and eICU-CRD, with higher macro AUROC, AUPRC, and F1 than the corresponding baselines in each case. Our findings show that restructuring existing EHR data can improve clinical prediction without changing the model or collecting additional data.

1 INTRODUCTION

Electronic health records (EHRs) represent medical histories as trajectories of diagnoses, procedures, medications, laboratory measurements, and other clinical events. These histories vary widely in length and detail, reflecting differences in both patient care and how that care is recorded. EHR foundation models (EHR-FMs) aim to learn from these longitudinal records and support multiple clinical prediction tasks. Autoregressive EHR-FMs represent each trajectory as a sequence of tokens and learn to predict the next token from those preceding it, as shown in Figure 1. Their architectures and training pipelines often draw on language modeling frameworks, with medical event tokens taking the place of text tokens (Renc et al., 2024; Kraljevic et al., 2024; Fallahpour et al., 2025). For zero-shot clinical prediction, a model is prompted with a patient's observed history and stochastically generates multiple future *rollouts*. The fraction of rollouts containing an outcome provides an estimate of that patient's risk of that outcome (Renc et al., 2025). Pretraining therefore directly shapes how these models perform on downstream clinical prediction tasks.

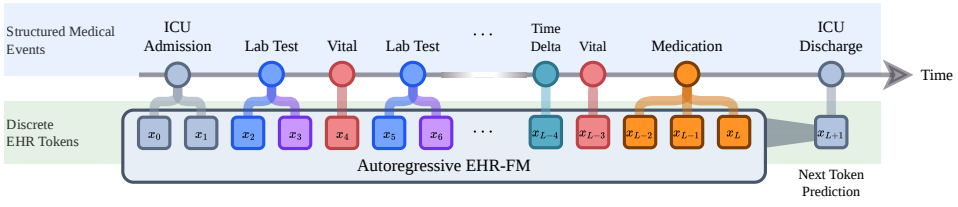
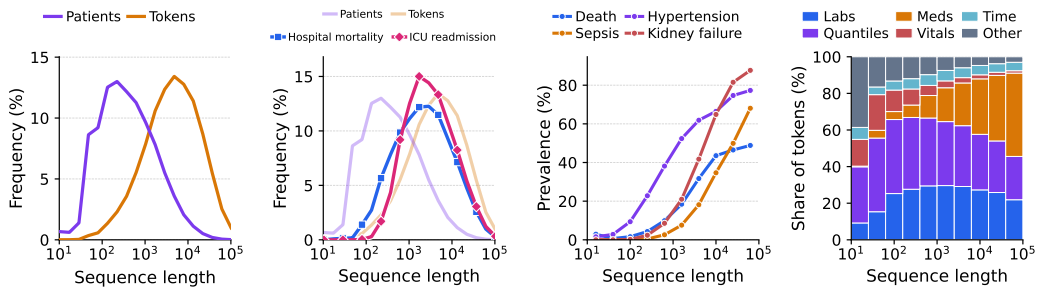


Figure 1: Autoregressive prediction from a patient trajectory. Medical events are encoded as discrete tokens, and the EHR-FM predicts the next token from the preceding clinical context.

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



(a) Patients and tokens. (b) Clinical task instances. (c) Outcome prevalence. (d) Token composition.

Figure 2: Patient trajectory length and clinical content in MIMIC-IV v2.2. Length is the number of tokens in a patient's complete trajectory. (a) Patient and token distributions by length. (b) Corresponding distributions of hospital mortality and ICU readmission prediction instances. (c) Recorded outcome prevalence by length. (d) Token composition by length.

How patient trajectories are assembled into pretraining sequences determines how much each patient contributes to learning and which events are available as context. This choice is particularly consequential for zero-shot EHR-FMs, which make clinical predictions without task-specific fine-tuning. A common approach concatenates patient trajectories into a global token stream and samples fixed-length windows from it (Renc et al., 2024; 2025). These windows can cross patient boundaries, allowing tokens from one patient to attend to another patient's events under standard causal attention. This introduces unrelated context of the kind found to distract language models trained on packed documents (Zhao et al., 2024).

In MIMIC-IV v2.2 (Johnson et al., 2023a;b), patient trajectories span several orders of magnitude in length. Sampling global-stream windows gives longer histories proportionally more next-token targets, so exposure follows the token rather than patient distribution (Figure 2a). Clinical prediction instances, such as hospital admissions and ICU discharges, follow a different length distribution from either (Figure 2b). Clinical content also varies with length: recorded outcome prevalence rises in longer trajectories (Figure 2c), while the mix of medical event tokens changes (Figure 2d). Reweighting trajectories therefore changes both patient exposure and the clinical information encountered during pretraining. We evaluate multiple outcomes to assess whether these effects extend beyond individual tasks.

We study sequence composition under a fixed model architecture and training recipe. We compare Global Stream (GS) with a patient-boundary-masked version (GS+M) that uses the same windows but prevents attention across patients. We then introduce Patient Sampling (PS), Recursive Patient Sampling (RPS), and Ordered Global Stream (OGS), which control patient contributions through different ways of constructing pretraining sequences. Figure 3 illustrates how GS and OGS give patients with unequal trajectory lengths different expected contributions to pretraining. These comparisons let us examine cross-patient attention and sequence composition as distinct factors in zero-shot clinical prediction.

Across four MIMIC-IV v2.2 tasks and five seeds, OGS achieves the highest mean performance on macro AUROC, AUPRC, F1, and balanced accuracy. Relative to GS, OGS improves macro AUPRC by 2.7%, F1 by 9.3%, and balanced accuracy by 3.7%. We further evaluate OGS in four additional dataset and tokenization settings, including INSPIRE and eICU-CRD. Relative to GS+M, OGS improves mean macro AUROC, AUPRC, and F1 across these settings by 2.0%, 2.3%, and 8.2%, respectively, with gains in all three metrics in every setting. These findings identify pretraining sequence composition as an important design choice for EHR foundation models.

2 Related Work

2.1 Autoregressive EHR-FMs

Autoregressive models learn from longitudinal EHRs by predicting future events from a patient's recorded history. CLMBR showed that next-event prediction can yield transferable patient rep-

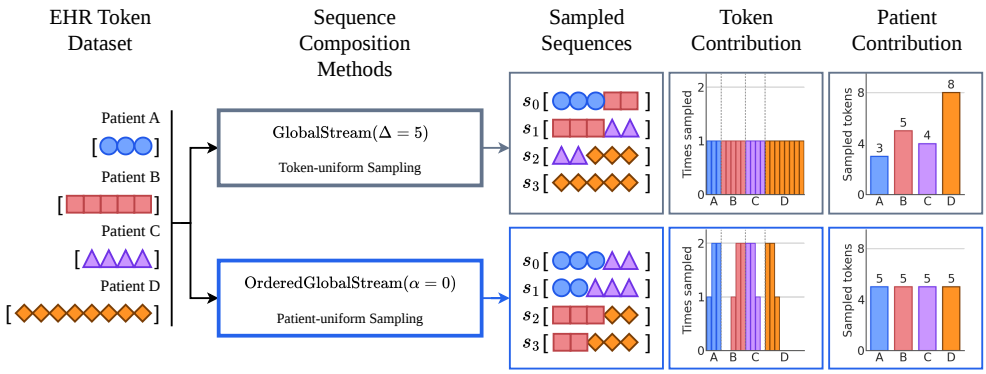


Figure 3: Sequence composition for four patients with unequal trajectory lengths. In the Global Stream example, each token appears once, so longer trajectories contribute more tokens. Ordered Global Stream ($\alpha = 0$) samples windows repeatedly to give patients approximately equal expected contributions.

resentations, particularly when labelled data are scarce (Steinberg et al., 2021). EHRSHOT subsequently provided a benchmark for evaluating pretrained EHR models across clinical prediction tasks (Wornow et al., 2023). Foresight and ETHOS extend event prediction to generative forecasting of patient trajectories (Kraljevic et al., 2024; Renc et al., 2024). ETHOS pretrains on concatenated patient timelines and supports rollout-based zero-shot clinical prediction (Renc et al., 2024), the evaluation protocol we adopt here. Other work has examined architectures that can process longer histories, the effect of context length, and scaling with compute, model size, and data volume (Fallahpour et al., 2025; Wornow et al., 2025; Zhang et al., 2025; Waxler et al., 2025).

2.2 Sequence Composition and Document Packing

Language-model pretraining often packs documents into fixed-length sequences to use the available context efficiently (Ding et al., 2024). Packing determines both which documents appear together and whether tokens can use preceding documents as context. With standard causal attention, a token can attend to earlier tokens from an unrelated document in the same sequence. Zhao et al. (2024) study this cross-document attention and show that restricting attention to tokens within each document can improve downstream behaviour. Changing the contents of a sequence offers another way to shape context. Shi et al. (2024) find that grouping semantically related documents improves performance relative to random packing. These results distinguish two decisions that can also arise with patient trajectories: how histories are placed into sequences, and whether attention crosses their boundaries.

2.3 Sampling and Reweighting

Sampling and reweighting change how often different parts of a dataset contribute to the training objective. DoReMi learns a mixture over language-model pretraining domains using a proxy model, without selecting weights against particular downstream tasks (Xie et al., 2023). Its results show that training exposure can affect downstream performance even when the underlying data sources remain the same. Clinical prediction work has likewise used patient reweighting to address temporal and cross-institutional shifts, and downsampling or sample weighting to address rare outcomes (Lee et al., 2023; Zhu et al., 2025; Yuan et al., 2022). In autoregressive EHR pretraining, sampling determines how frequently each patient's tokens provide prediction targets. Taken together, this work motivates studying patient exposure and training context as linked consequences of sequence composition.

Algorithm 1 Global Stream (GS)

Require: Ordered patient timelines $\mathcal{D} = (\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)})$, context length L, stride Δ

$\mathbf{g} \leftarrow \mathbf{x}^{(1)} \| \cdots \| \mathbf{x}^{(N)}$

$K \leftarrow 1 + \left\lfloor \frac{|\mathbf{g}|-L-1}{\Delta} \right\rfloor$ ▷ Number of valid windows

$k \sim \mathrm{Uniform}\{0, \ldots, K-1\}$ ▷ Sample a window index

$j \leftarrow 1 + k\Delta$

$\mathbf{s} \leftarrow (g_j, \ldots, g_{j+L})$ ▷ Extract an $(L+1)$-token window

return $\mathbf{s}$

Algorithm 2 Patient Sampling (PS)

Require: Patient timelines $\mathcal{D} = (\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)})$, context length L, patient sampling distribution p_α

$i \sim \mathrm{Categorical}(p_\alpha)$ ▷ Sample a patient

$T_i \leftarrow |\mathbf{x}^{(i)}|$

$j \sim \mathrm{Uniform}\{2-L, \ldots, T_i-1\}$ ▷ Allow partial windows

$\mathbf{s} \leftarrow \left(x^{(i)}_{\max(1,j)}, \ldots, x^{(i)}_{\min(T_i, j+L)}\right)$ ▷ Clip to the patient timeline

$\mathbf{s} \leftarrow \textsc{PadRight}(\mathbf{s}, L+1)$

return $\mathbf{s}$

3 Sequence Composition Methods

We pretrain on a tokenized EHR dataset $\mathcal{D} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}\}$ containing N patients. Here, $\mathbf{x}^{(i)} = (x_1^{(i)}, \ldots, x_{T_i}^{(i)})$ is the chronologically ordered token sequence for patient i, with length T_i. Given a context length L, each sequence-composition method constructs a training sequence $\mathbf{s} = (s_1, \ldots, s_{L+1})$ from $\mathcal{D}$, padding any unused positions. The model receives $\mathbf{s}_{1:L}$ as input and predicts the shifted targets $\mathbf{s}_{2:L+1}$.

3.1 Global Stream

We use Global Stream (GS) as our baseline, following the concatenated-timeline pretraining approach used by ETHOS (Renc et al., 2024). GS concatenates patient trajectories into a token stream $\mathbf{g}$ and defines candidate windows of $L+1$ tokens at starting positions separated by stride Δ (Algorithm 1). During training, these windows are sampled uniformly with replacement. Longer trajectories occupy more positions in $\mathbf{g}$ and therefore contribute approximately proportionally more next-token training targets. Windows may span patient boundaries, allowing tokens to attend to preceding patients under standard causal attention. To assess the effect of cross-patient attention, we also evaluate Global Stream with masking (GS+M). It samples the same windows as GS but prevents attention across patient boundaries. The methods we introduce below also prevent cross-patient attention, so GS+M lets us assess the effects of sequence composition separately from those of attention masking.

3.2 Patient Contribution Weighting

We control each patient's contribution to pretraining through their share of next-token prediction targets. Let M_i be the number of loss-eligible next-token targets in patient i's trajectory. For $\alpha \in [0, 1]$, we define the target share as

$$\pi_i(\alpha) = \frac{M_i^\alpha}{\sum_{j=1}^{N} M_j^\alpha} = \mathrm{softmax}_i(\alpha \log M_i). \tag{1}$$

At $\alpha = 0$, every patient has an equal target share; at $\alpha = 1$, shares are proportional to M_i, approximately matching the token-proportional exposure of GS. Intermediate values shift exposure between these endpoints, as illustrated for Patient Sampling in Appendix Figure 8. The methods below use different sampling procedures to approach these target shares.

Algorithm 3 Recursive Patient Sampling (RPS)

Require: Patient timelines $\mathcal{D}$, context length L, patient sampling distribution p_α

$\mathbf{s} \leftarrow ()$
$R \leftarrow L + 1$ ▷ Remaining capacity
while $R \geq 2$ **do**
  $i \sim \text{Categorical}(p_\alpha)$
  $j \sim \text{Uniform}\{3 - R, \ldots, T_i - 1\}$ ▷ Allow partial chunks near either end
  $\mathbf{c} \leftarrow \left(x^{(i)}_{\max(1,j)}, \ldots, x^{(i)}_{\min(T_i, j+R-1)}\right)$ ▷ Keep tokens within the trajectory
  $\mathbf{s} \leftarrow \mathbf{s} \| \mathbf{c}$
  $R \leftarrow R - |\mathbf{c}|$
end while
$\mathbf{s} \leftarrow \text{PADRIGHT}(\mathbf{s}, L + 1)$
return $\mathbf{s}$

Algorithm 4 Ordered Global Stream (OGS)

Require: Patient timelines $\mathcal{D}$, context length L, stride Δ, exponent α

$\boldsymbol{\rho} \leftarrow \text{argsort}_i(T_i)$ ▷ Order patients by length
$\mathbf{g} \leftarrow \mathbf{x}^{(\rho_1)} \| \cdots \| \mathbf{x}^{(\rho_N)}$ ▷ Concatenate in this order
Let $a(k)$ be the patient at position k in $\mathbf{g}$.
$\mathcal{I} \leftarrow \{1 + m\Delta : m \geq 0,\ 1 + m\Delta + L \leq |\mathbf{g}|\}$ ▷ Valid window starts
$q_k \leftarrow M^\alpha_{a(k)} / T_{a(k)}$ for $k = 1, \ldots, |\mathbf{g}|$ ▷ Distribute patient weight across tokens
$w_j \leftarrow \frac{1}{L+1} \sum_{k=j}^{j+L} q_k$ for $j \in \mathcal{I}$ ▷ Mean weight in window
$p_j \leftarrow w_j \Big/ \sum_{r \in \mathcal{I}} w_r$ for $j \in \mathcal{I}$ ▷ Normalize window weights
$j \sim \text{Categorical}(\{p_r\}_{r \in \mathcal{I}})$ ▷ Sample a window
return $(g_j, \ldots, g_{j+L})$

3.3 Patient Sampling

Patient Sampling (PS) first selects a patient, then samples a contiguous window from that patient's trajectory (Algorithm 2). To include targets near either end, the sampled start may precede the first token, or the window may extend beyond the last. We retain the tokens within the trajectory and pad the sequence on the right to length $L + 1$. PS never combines tokens from different patients in the same training sequence. Windows from short trajectories or near their boundaries contain fewer prediction targets. Let E_i be the expected number of targets in a window sampled from patient i. We account for this variation by drawing patient i with probability $p_\alpha(i) \propto \pi_i(\alpha)/E_i$, aiming to match the desired target shares. The normalized probabilities and their derivation are given in Appendix A.

3.4 Recursive Patient Sampling

Patient Sampling can leave much of a sequence padded when a patient's trajectory is short. Recursive Patient Sampling (RPS) fills the unused space with additional patient chunks (Algorithm 3). It repeatedly draws a patient, samples a contiguous chunk that fits the remaining capacity, and appends it to the sequence. The process stops when fewer than two positions remain, and any unused position is padded. Chunks may come from different patients or from noncontiguous parts of the same trajectory. We mask the next-token target at each join so the model is not trained to predict across unrelated chunks. We also mask attention across all chunk boundaries. RPS aims for the same target shares $\pi_i(\alpha)$ as PS, but its patient-draw weights differ because the capacity available for each chunk changes during packing. The weighting procedure is described in Appendix A.2.

3.5 Ordered Global Stream

Ordered Global Stream (OGS) aims to control patient contributions while retaining GS's full-length windows, without the padding of PS or the iterative packing of RPS (Algorithm 4). We sort patients by trajectory length before concatenating their timelines, placing patients of similar length near one

Table 1: Dataset and tokenization statistics. Token counts and trajectory lengths are calculated after tokenization. The final column gives the percentage of patients whose complete trajectory exceeds 2,048 tokens, which is the size of the context window, L, we use in our experiments.

Dataset	Tokenization	Patients	Tokens	Vocab. size	Median length (tokens)	Length $> 2{,}048$ (% of patients)
MIMIC-IV v2.2	ETHOS	285,606	362,697,213	4,509	338	14.2%
MIMIC-IV v3.1	ETHOS	346,262	500,700,081	4,598	365	16.1%
MIMIC-IV v2.2	Ours	287,511	576,657,770	6,431	512	22.2%
INSPIRE	Ours	99,900	176,560,774	1,313	840	18.3%
eICU-CRD	Ours	138,963	461,065,895	8,000	1,719	43.1%

another. Windows that cross patient boundaries therefore tend to contain tokens with similar sampling weights. As in GS+M, we mask attention across patient boundaries within sampled windows.

We assign each token belonging to patient i a weight of M_i^α / T_i, distributing that patient's weight across their trajectory. The weights of patient i's tokens sum to M_i^α. Each candidate window receives the mean weight of its $L+1$ tokens, and we sample windows in proportion to these weights. At $\alpha = 0$, patients have equal total token weight; at $\alpha = 1$, token weights are approximately uniform for longer trajectories, making window sampling similar to GS. Because weights are assigned to windows rather than individual prediction targets, the resulting patient contributions approximate $\pi_i(\alpha)$. Appendix A.3 provides the weighting details.

4 Experimental Setup

4.1 Datasets

Our primary experiments use MIMIC-IV v2.2 (Johnson et al., 2023a;b), a widely used EHR dataset, to compare sequence-composition methods and patient-weighting settings. Our MIMIC-IV data include hospital and intensive care records from Beth Israel Deaconess Medical Center in Boston, together with linked emergency department records. We assess generalization using three additional datasets. MIMIC-IV v3.1 (Johnson et al., 2024; 2023b) is a later release of the same data with additional patient records. eICU-CRD v2.0 (Pollard et al., 2019; 2018) contains intensive care records from multiple hospitals across the United States. INSPIRE v1.4.2 (Lim & Lee, 2026; Lim et al., 2024) contains perioperative records from Seoul National University Hospital in South Korea. All datasets were obtained through PhysioNet (Pollard et al., 2026). We split each dataset by patient into 80% training, 10% validation, and 10% test sets. We train a new model from scratch for each experiment using the corresponding training split. Table 1 summarizes the datasets used in our experiments.

4.2 Tokenization

ETHOS tokenization. For the primary MIMIC-IV v2.2 experiments, we tokenize patient records using the code released with ETHOS (Renc et al., 2025). We apply the same preprocessing and tokenization configuration to MIMIC-IV v3.1. For both releases, we include the hospital (`hosp`), intensive care unit (`icu`), and emergency department (`ed`) modules. The resulting trajectories encode demographics, clinical events, laboratory values, and elapsed time.

Unified EHR tokenization. We develop a pipeline for MIMIC-IV v2.2, eICU-CRD, and INSPIRE, converting source records to Medical Event Data Standard (MEDS) events (Arnrich et al., 2024) through dataset-specific adapters and applying a shared tokenization configuration. Clinical codes, measurements, units, and elapsed time are represented as discrete tokens, with numeric measurements assigned to 10 quantile bins. An algorithm inspired by Byte-Pair Encoding (Sennrich et al., 2016) merges frequent adjacent token pairs. For both pipelines, we fit the tokenizer separately on each dataset's training split and apply it to the corresponding validation and test splits. Appendix D details the encoding and merging procedures and compares the two pipelines.

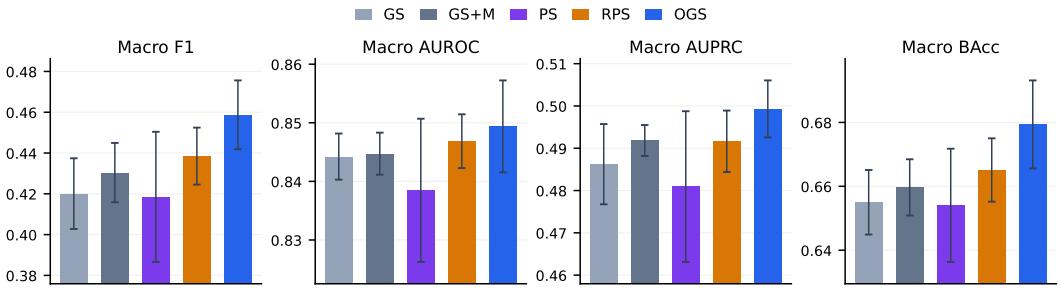


Figure 4: Macro downstream performance across four clinical prediction tasks on MIMIC-IV v2.2 test set, reported as mean $\pm$ standard deviation over five seeds.

4.3 Model and Training Recipe

All experiments use a GPT-2-style decoder-only Transformer (Radford et al., 2019) with six layers, a hidden dimension of 768, 12 attention heads, and a context length of $L = 2048$. Each model has approximately 50M parameters, with the exact count varying by vocabulary size. The loss is averaged over all loss-eligible targets in each batch. We hold the number of optimisation steps, batch size, and context length fixed across methods; the number of loss-eligible targets varies with sequence composition. Appendix Figure 9 compares expected training-data coverage under a fixed window budget. Padding and demographic tokens are excluded from the loss. Because each patient sequence or chunk begins with demographic tokens, prediction targets at their boundaries are also excluded. For RPS, we also mask targets at chunk joins. GS uses standard causal attention. GS+M, RPS, and OGS prevent attention across patient boundaries using the FlashAttention masking implementation of Zhao et al. (2024).

We train each model for 500,000 optimization steps on a single NVIDIA H100 GPU. We select the checkpoint with the lowest next-token validation loss. Validation uses global-stream windows from the corresponding validation split, with standard causal attention for GS and patient-boundary masking for all other methods. For the primary comparison, we use the same five random seeds for each method and report the mean and standard deviation. Table 2 describes the training recipe, with further details in Appendix B.

4.4 Evaluation Protocol

Zero-shot clinical prediction. We evaluate pretrained models without task-specific fine-tuning. For each task, we prompt the model with a patient's trajectory up to the prediction time and generate 20 independent rollouts at temperature 1. Generation stops when a task-specific stopping condition is met or the maximum number of steps is reached. Rollouts that reach the step limit without producing a valid terminating token are disregarded. We estimate risk as the fraction of completed rollouts that reach the positive outcome.

Clinical tasks. The primary MIMIC-IV v2.2 evaluation covers hospital mortality, ICU mortality, ICU readmission, and ICU admission. The mortality tasks predict death during the corresponding stay. ICU readmission predicts a further ICU stay after ICU discharge, while ICU admission predicts entry to the ICU after hospital admission. For the other datasets, we evaluate tasks supported by their recorded events. We report F1, AUROC, AUPRC, and balanced accuracy (BAcc) for each task, using a risk threshold of 0.5 for F1 and BAcc. Macro scores are unweighted means across the tasks in each dataset and tokenization setting. Appendix E specifies the full evaluation protocol and task definitions across all datasets.

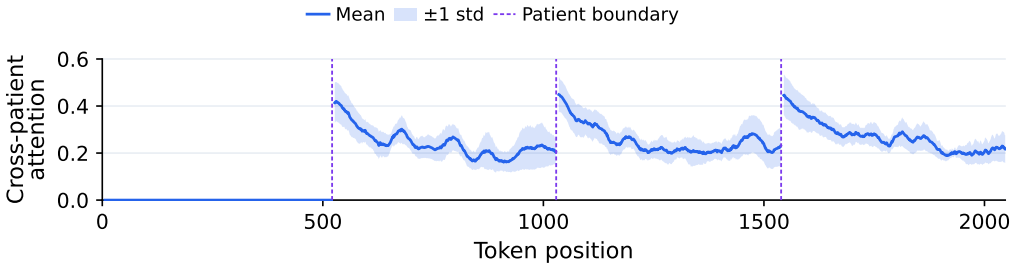


Figure 5: Cross-patient attention in an example GS sequence. The line and shading show mean attention to preceding patients and ±1 standard deviation; dotted lines mark patient boundaries.

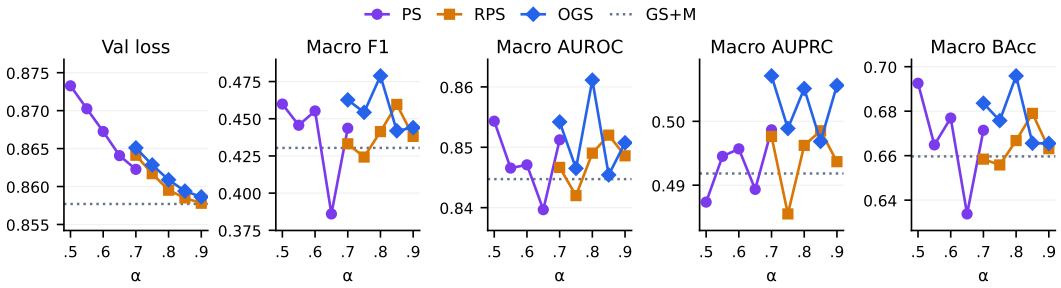


Figure 6: Local patient-weighting sweeps on MIMIC-IV v2.2. Panels show next-token validation loss and macro validation performance across four clinical tasks as α varies. Dotted lines indicate the GS+M baseline.

5 Results and Analysis

5.1 Sequence Composition on MIMIC-IV

We compare PS, RPS, and OGS with the GS and GS+M baselines on MIMIC-IV v2.2 using ETHOS tokenization. We select α separately for each proposed method through coarse and local sweeps, maximizing the mean of macro AUROC and macro AUPRC on the validation set. This yields $\alpha = 0.7$ for PS, 0.85 for RPS, and 0.8 for OGS. Figure 4 reports zero-shot performance across four clinical tasks as the mean and standard deviation over five seeds. Masking attention across patient boundaries in GS produces small improvements across all four macro metrics. Figure 5 shows that tokens immediately following a patient boundary assign substantial attention to preceding patients when this mask is absent. These tokens belong to unrelated clinical histories, yet the GS model uses them as context.

Among the proposed methods, RPS improves most macro metrics over the baselines, whereas PS has lower mean performance and greater variation across seeds. OGS achieves the highest mean on all four metrics, including when compared with GS+M, which uses the same patient-boundary attention mask. Relative to GS, OGS improves macro F1, AUROC, AUPRC, and balanced accuracy by 3.9, 0.5, 1.3, and 2.4 percentage points, respectively. These correspond to relative gains of 9.3%, 0.6%, 2.7%, and 3.7%.

5.2 Patient Weighting and Validation Loss

Figure 6 shows performance near the selected α values; Tables 3 and 4 report the full initial and local sweeps. Next-token validation loss decreases as α increases, reaching its minimum at $\alpha = 1$ for all three methods. However, macro AUROC and AUPRC vary across the local sweep rather than improving steadily with validation loss. For OGS, $\alpha = 0.8$ yields macro AUROC and AUPRC of 0.861 and 0.505, compared with 0.834 and 0.481 at $\alpha = 1$, despite the latter's lower validation loss. These results show that lower next-token validation loss does not necessarily translate into better downstream clinical performance.

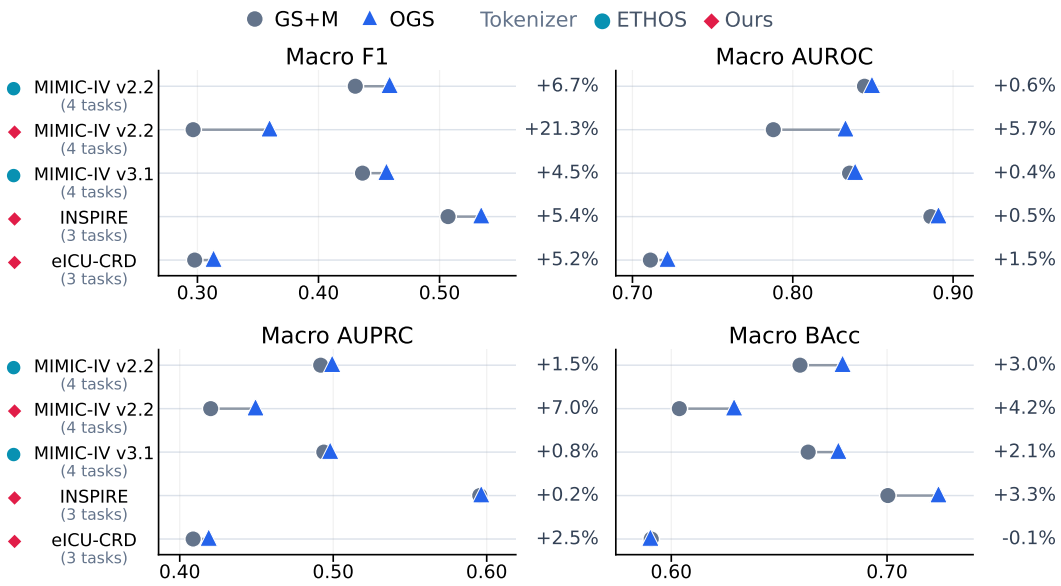


Figure 7: Generalization across datasets and tokenizations. Points show macro test-set performance across the indicated tasks, averaged over five seeds for MIMIC-IV v2.2 with ETHOS tokenization and reported for one seed in each additional setting. Annotations report relative changes from GS+M to OGS. Markers beside dataset labels identify tokenizers. Horizontal scales differ between metrics.

5.3 Generalization Across Datasets and Tokenizations

We compare OGS ($\alpha = 0.8$) with GS+M in four additional experiments: MIMIC-IV v2.2 with our unified tokenizer, MIMIC-IV v3.1 with ETHOS tokenization, and INSPIRE and eICU-CRD with our unified tokenizer. In each case, we train both models from scratch with the same configuration and patient-boundary mask. Figure 7 shows the four additional experiments alongside the primary MIMIC-IV v2.2 result; Table 10 reports task-level scores. Due to computational cost, we use one seed per method in each additional experiment. In the five-seed primary comparison, OGS improves macro AUROC, AUPRC, and F1 over GS+M by 0.6%, 1.5%, and 6.7%, respectively. Across the four additional experiments, OGS's average macro scores improve by 2.0%, 2.3%, and 8.2%, respectively. All three of these metrics improve across all four generalization experiments; balanced accuracy improves in three of four. The gains vary by task: on eICU-CRD, ICU-readmission F1 falls from 0.411 to 0.343 despite gains on the other two tasks.

6 Conclusion

Pretraining sequence composition shapes patient contributions to learning and the clinical context available for prediction. Across four MIMIC-IV v2.2 tasks, OGS achieved the highest mean performance on all four macro metrics, including relative improvements of 9.3% in macro F1, 2.7% in AUPRC, and 3.7% in balanced accuracy over GS. The improvement over GS+M shows that sequence composition improves clinical prediction even when attention is already restricted to individual patients. Using $\alpha = 0.8$ selected on MIMIC-IV v2.2, we evaluated OGS across additional datasets and tokenization pipelines without retuning patient weighting. OGS improved macro AUROC, AUPRC, and F1 over GS+M in all four additional experiments, with relative gains in average macro scores of 2.0%, 2.3%, and 8.2%, respectively. Evaluation was limited to one architecture and training recipe; future work should assess whether these gains persist across model configurations and other clinical settings. These results support sequence composition as a practical design choice for EHR foundation models: reorganizing existing patient records can improve zero-shot clinical prediction without changing the model architecture.

AI USE STATEMENT

In this work, we used generative AI tools for the following:

- formulate mathematical claims
- design or provide feedback on research methodology or experiments
- implement methods
- clean and reformat dataset
- support qualitative and thematic data analysis

We have not used generative AI tools for the following:

- help develop theoretical models or conceptual frameworks
- provide critical ingredients for proving mathematical claims
- propose or refine hypotheses
- interpret results
- Generate synthetic data sets

, and the following:

- assist with translation
- assist in the writing of proofs

are not applicable to this work. Additionally, we used generative AI tools for the followings:

- create or modify scientific figures or images
- create or edit software code
- analyze existing literature

We have reviewed all AI-assisted work. Methods and core code related to the paper was review by at least 1 author and we verified all LLM-generated outputs that have been used in this work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Section 3 and Appendix A specify the sequence-composition algorithms and patient-weighting calculations. Section 4 reports the dataset versions and patient-level splits, tokenization procedures, model and training configuration, and rollout evaluation protocol, with further details in Appendices B, E, and F. Section 5 and the appendix tables report the patient-weighting sweeps, five-seed primary comparison, and task-level results across datasets. We identify the sources of all datasets used in Section 4.1.

REFERENCES

Bert Arnrich, Edward Choi, Jason Alan Fries, Matthew B.A. McDermott, Jungwoo Oh, Tom Pollard, Nigam Shah, Ethan Steinberg, Michael Wornow, and Robin van de Water. Medical event data standard (MEDS): Facilitating machine learning for health. In *ICLR 2024 Workshop on Learning from Time Series For Health*, 2024. URL https://openreview.net/forum?id=IsHy2ebjIG.

Hantian Ding, Zijian Wang, Giovanni Paolini, Varun Kumar, Anoop Deoras, Dan Roth, and Stefano Soatto. Fewer truncations improve language modeling. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 11030–11048. PMLR, July 2024.

Adibvafa Fallahpour, Mahshid Alinoori, Wenqian Ye, Xu Cao, Arash Afkanpour, and Amrit Krishnan. EHRMamba: Towards generalizable and scalable foundation models for electronic health records. In Stefan Hegselmann, Helen Zhou, Elizabeth Healey, Trenton Chang, Caleb Ellington, Vishwali Mhasawade, Sana Tonekaboni, Peniel Argaw, and Haoran Zhang (eds.), *Proceedings of the 4th Machine Learning for Health Symposium*, volume 259 of *Proceedings of Machine Learning Research*, pp. 291–307. PMLR, December 2025.

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. MIMIC-IV. *PhysioNet*, January 2023a. doi: 10.13026/6mm1-ek67. URL https://doi.org/10.13026/6mm1-ek67. Version 2.2.

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

Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, Li-wei H. Lehman, Leo A. Celi, and Roger G. Mark. Mimic-iv, a freely accessible electronic health record dataset. *Scientific Data*, 10(1):1, 2023b. doi: 10.1038/s41597-022-01899-x. URL https://doi.org/10.1038/s41597-022-01899-x.

Zeljko Kraljevic, Dan Bean, Anthony Shek, Rebecca Bendayan, Harry Hemingway, Joshua Au Yeung, Alexander Deng, Alfred Balston, Jack Ross, Esther Idowu, James T Teo, and Richard J B Dobson. Foresight—a generative pretrained transformer for modelling of patient timelines using electronic health records: A retrospective modelling study. *The Lancet Digital Health*, 6 (4):e281–e290, April 2024. ISSN 2589-7500. doi: 10.1016/S2589-7500(24)00025-6.

Seungyeon Lee, Changchang Yin, and Ping Zhang. Stable clinical risk prediction against distribution shift in electronic health records. *Patterns*, 4(9), September 2023. ISSN 2666-3899. doi: 10.1016/j.patter.2023.100828.

Leerang Lim and Hyung-Chul Lee. INSPIRE, a publicly available research dataset for perioperative medicine. *PhysioNet*, June 2026. doi: 10.13026/1eay-yc85. URL https://doi.org/10.13026/1eay-yc85. Version 1.4.2.

Leerang Lim, Hyeonhoon Lee, Chul-Woo Jung, Dayeon Sim, Xavier Borrat, Tom J. Pollard, Leo A. Celi, Roger G. Mark, Simon T. Vistisen, and Hyung-Chul Lee. Inspire, a publicly available research dataset for perioperative medicine. *Scientific Data*, 11(1):655, 2024. doi: 10.1038/s41597-024-03517-4. URL https://doi.org/10.1038/s41597-024-03517-4.

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

Tom Pollard, Benjamin E. Moody, Li-wei H. Lehman, Brian J. Gow, Chrystinne Fernandes, Chen Xie, Alistair Johnson, Roger G. Mark, and Thomas Heldt. PhysioNet as a global platform for biomedical research. *Nature Health*, 1(8):792–795, August 2026. ISSN 3005-0693. doi: 10.1038/s44360-026-00096-z. URL https://doi.org/10.1038/s44360-026-00096-z.

Tom J. Pollard, Alistair E. W. Johnson, Jesse D. Raffa, Leo A. Celi, Roger G. Mark, and Omar Badawi. The eicu collaborative research database, a freely available multi-center database for critical care research. *Scientific Data*, 5(1):180178, 2018. doi: 10.1038/sdata.2018.178. URL https://doi.org/10.1038/sdata.2018.178.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

Pawel Renc, Yugang Jia, Anthony E. Samir, Jaroslaw Was, Quanzheng Li, David W. Bates, and Arkadiusz Sitek. Zero shot health trajectory prediction using transformer. *npj Digital Medicine*, 7(1):256, September 2024. ISSN 2398-6352. doi: 10.1038/s41746-024-01235-0.

Pawel Renc, Michal K Grzeszczyk, Nassim Oufattole, Deirdre Goode, Yugang Jia, Szymon Bieganski, Matthew B A McDermott, Jaroslaw Was, Anthony E Samir, Jonathan W Cunningham, David W Bates, and Arkadiusz Sitek. Foundation model of electronic medical records for adaptive risk estimation. *GigaScience*, 14:giaf107, January 2025. ISSN 2047-217X. doi: 10.1093/gigascience/giaf107.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In Katrin Erk and Noah A. Smith (eds.), *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1162. URL https://aclanthology.org/P16-1162/.

Weijia Shi, Sewon Min, Maria Lomeli, Chunting Zhou, Margaret Li, Xi Victoria Lin, Noah A. Smith, Luke Zettlemoyer, Wen-tau Yih, and Mike Lewis. In-context pretraining: Language modeling beyond document boundaries. In *The Twelfth International Conference on Learning Representations*, 2024.

Ethan Steinberg, Ken Jung, Jason A. Fries, Conor K. Corbin, Stephen R. Pfohl, and Nigam H. Shah. Language models are an effective representation learning technique for electronic health record data. *Journal of Biomedical Informatics*, 113:103637, 2021. ISSN 1532-0464. doi: 10.1016/j.jbi.2020.103637.

Shane Waxler, Paul Blazek, Davis White, Daniel Sneider, Kevin Chung, Mani Nagarathnam, Patrick Williams, Hank Voeller, Karen Wong, Matthew Swanhorst, Sheng Zhang, Naoto Usuyama, Cliff Wong, Tristan Naumann, Hoifung Poon, Andrew Loza, Daniella Meeker, Seth Hain, and Rahul Shah. Generative medical event models improve with scale. August 2025.

Michael Wornow, Rahul Thapa, Ethan Steinberg, Jason Fries, and Nigam Shah. EHRSHOT: An EHR benchmark for few-shot evaluation of foundation models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 67125–67137. Curran Associates, Inc., 2023. doi: 10.52202/075280-2933.

Michael Wornow, Suhana Bedi, Miguel Angel Fuentes Hernandez, Ethan Steinberg, Jason Fries, Christopher Re, Sanmi Koyejo, and Nigam Shah. Context clues: Evaluating long context models for clinical prediction tasks on EHR data. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 16518–16555, 2025.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. DoReMi: Optimizing data mixtures speeds up language model pretraining. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, Nips '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Han Yuan, Feng Xie, Marcus Eng Hock Ong, Yilin Ning, Marcel Lucas Chee, Seyed Ehsan Saffari, Hairil Rizal Abdullah, Benjamin Alan Goldstein, Bibhas Chakraborty, and Nan Liu. AutoScore-Imbalance: An interpretable machine learning tool for development of clinical scores with rare events data. *Journal of Biomedical Informatics*, 129:104072, 2022. ISSN 1532-0464. doi: 10.1016/j.jbi.2022.104072.

Sheng Zhang, Qin Liu, Naoto Usuyama, Cliff Wong, Tristan Naumann, and Hoifung Poon. Exploring scaling laws for EHR foundation models. August 2025.

Yu Zhao, Yuanbin Qu, Konrad Staniszewski, Szymon Tworkowski, Wei Liu, Piotr Miłoś, Yuxiang Wu, and Pasquale Minervini. Analysing the impact of sequence composition on language model pre-training. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7897–7912, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.427.

He Zhu, Jun Bai, Na Li, Xiaoxiao Li, Dianbo Liu, David L. Buckeridge, and Yue Li. FedWeight: Mitigating covariate shift of federated learning on electronic health records data through patients re-weighting. *npj Digital Medicine*, 8(1):286, May 2025. ISSN 2398-6352. doi: 10.1038/s41746-025-01661-8.

A Patient Contribution Weights

Let T_i be the length of patient i's tokenized trajectory, and let M_i be its number of loss-eligible next-token targets. In our implementation, we exclude demographic tokens (such as age and gender) from our loss calculations following Renc et al. (2025). Padding tokens are also excluded from the loss. The weights aim to give patient i a share of training targets proportional to M_i^α. The counts below describe sampled tokens before demographic prefixes are added to the sequence.

A.1 Patient Sampling

PS draws a window start uniformly from $\{2-L,\dots,T_i-1\}$, giving $A_i = T_i + L - 2$ possible starts. Before demographic processing, each eligible target occurs in exactly L of these windows. The expected number of targets from a draw of patient i is therefore

$$E_i = \frac{LM_i}{T_i + L - 2}. \tag{2}$$

We divide the desired target weight by this expected count to obtain

$$p_\alpha(i) = \frac{M_i^\alpha / E_i}{\sum_j M_j^\alpha / E_j} = \mathrm{softmax}_i(\alpha \log M_i - \log E_i)\,. \tag{3}$$

Thus, before demographic processing, the expected target share is

$$\frac{p_\alpha(i)E_i}{\sum_j p_\alpha(j)E_j} = \frac{M_i^\alpha}{\sum_j M_j^\alpha} = \pi_i(\alpha). \tag{4}$$

This correction controls target exposure rather than the fraction of windows drawn from each patient.

A.2 Recursive Patient Sampling

RPS draws chunks with different remaining capacities. When R positions remain, patient i has $T_i + R - 3$ possible chunk starts, and each eligible target occurs in $R-1$ candidate chunks. Before demographic processing, the expected target count from that draw is

$$E_i(R) = \frac{(R-1)M_i}{T_i + R - 3}. \tag{5}$$

RPS uses one patient distribution across all capacities. We estimate how often each capacity occurs by simulating 4,096 packed sequences. If $c(R)$ is the resulting count, we set

$$\widehat{E}_i = \sum_{R=2}^{L+1} c(R)E_i(R), \qquad p_\alpha(i) = \frac{M_i^\alpha / \widehat{E}_i}{\sum_j M_j^\alpha / \widehat{E}_j}. \tag{6}$$

Starting from the PS weights, we simulate capacities and update these weights three times. This is an approximate correction because the capacity distribution depends on the weights and demographic processing can change chunk lengths.

A.3 Ordered Global Stream

OGS distributes each patient's weight across their tokens. If $a(k)$ is the patient at stream position k, the token weight is $q_k = M_{a(k)}^\alpha / T_{a(k)}$. The weights of patient i's tokens therefore sum to M_i^α. For each valid window start j, OGS uses the mean token weight

$$w_j = \frac{1}{L+1}\sum_{k=j}^{j+L} q_k, \qquad p_j = \frac{w_j}{\sum_{r\in\mathcal{I}} w_r}. \tag{7}$$

Sampling windows according to p_j shifts exposure toward the desired patient shares. The shares are approximate because overlapping windows can contain different numbers of eligible targets from each patient, particularly near boundaries.

A.4 Training Data Coverage

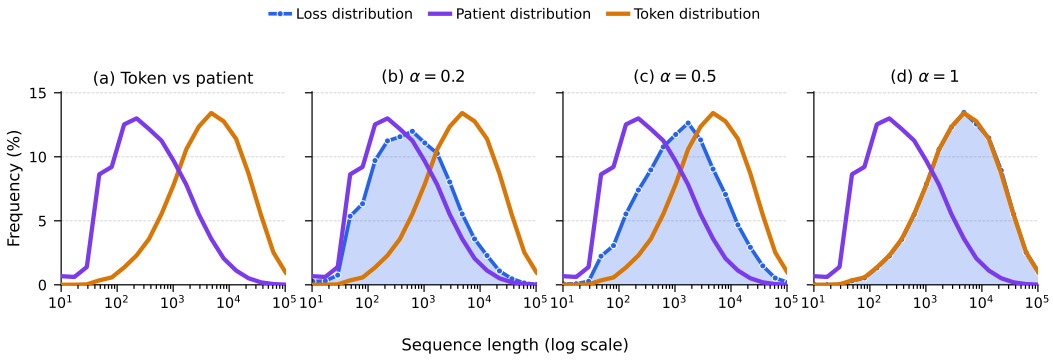


Figure 8: Expected training-target exposure by patient trajectory length under Patient Sampling. (a) Distributions of patients and tokens. (b–d) Expected target shares for $\alpha = 0.2$, 0.5, and 1. Each distribution sums to 100% across length bins.

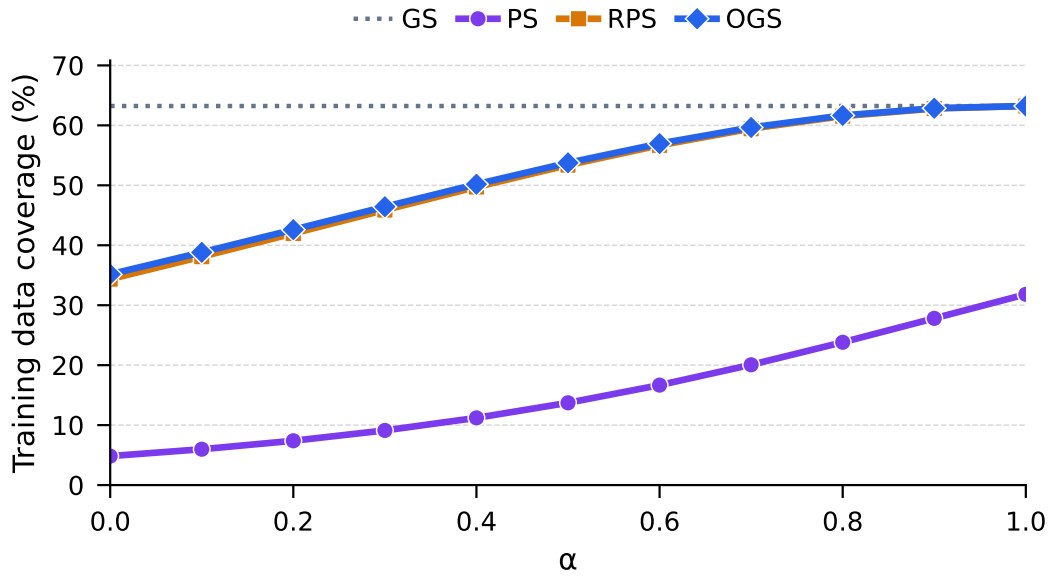


Figure 9: Expected training-data coverage on MIMIC-IV v2.2 as a function of α. Let u_k be the probability that corpus token position k appears in one sampled window. After B independent draws, its probability of being encountered at least once is $1 - (1 - u_k)^B$; coverage averages this quantity over all corpus positions, regardless of loss eligibility. We use $B \approx 141{,}593$ windows with context length 2,048, giving a nominal token budget equal to the training-corpus size. The dotted line shows GS, which is independent of α.

B TRAINING RECIPE

Table 2: Model architecture and pretraining configuration. The parameter count is for MIMIC-IV v2.2 with ETHOS tokenization; it varies with vocabulary size in the other settings.

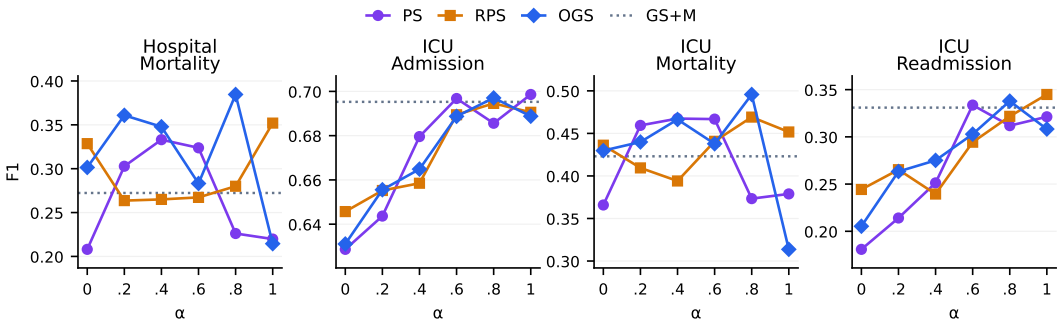
Parameter	Value
Architecture	GPT-2-style decoder (Radford et al., 2019)
Parameters	49,454,592
Transformer layers	6
Hidden dimension	768
Attention heads	12
Feed-forward dimension	3,072
Positional encoding	Rotary (base 10,000)
Context length L	2,048
Dropout	0.1
Batch size	128
Optimiser	AdamW
AdamW β_1, β_2	$0.9,\ 0.95$
Weight decay	0.1
Peak learning rate	6×10^{-4}
Learning-rate warmup	4,000 steps
Learning rate held at peak	136,000 steps
Linear learning-rate decay	260,000 steps
Final learning rate	6×10^{-5}
Training steps	500,000
Validation interval	10,000 steps
Training precision	BFloat16 autocast

The models use untied input and output token embeddings. We train from scratch on a single GPU for 500,000 optimiser steps, without early stopping. The learning rate increases linearly during warmup, remains at its peak for 136,000 steps, then decreases linearly to one tenth of the peak over 260,000 steps. It remains at that value for the final 100,000 steps. The training loss is the mean cross-entropy over loss-eligible next-token targets.

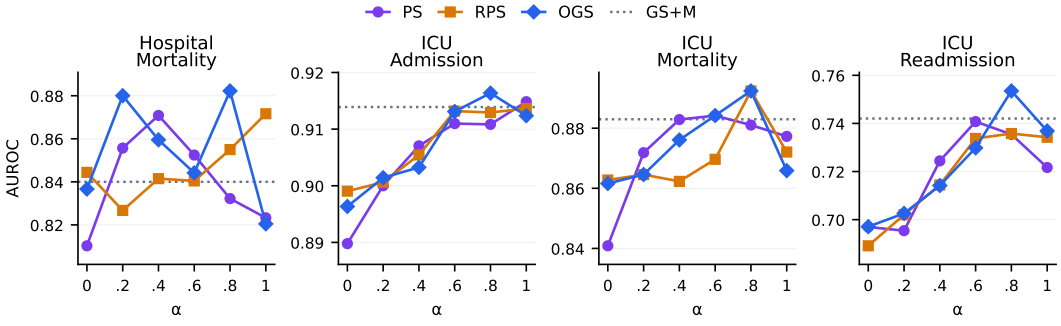
For GS and GS+M, every valid global-stream window start is a training candidate (stride 1); OGS uses candidate starts separated by 64 tokens. All methods use the `keep_left_most` demographic strategy which moves all demographic tokens into the left most positions of that patients sequence. Validation uses fixed windows from the concatenated validation trajectories with stride 2,048. We evaluate every 10,000 steps and select the checkpoint with the lowest token-mean validation loss. GS uses standard causal attention; the other methods restrict attention at patient or chunk boundaries, as described in Section 3.

C RESULTS

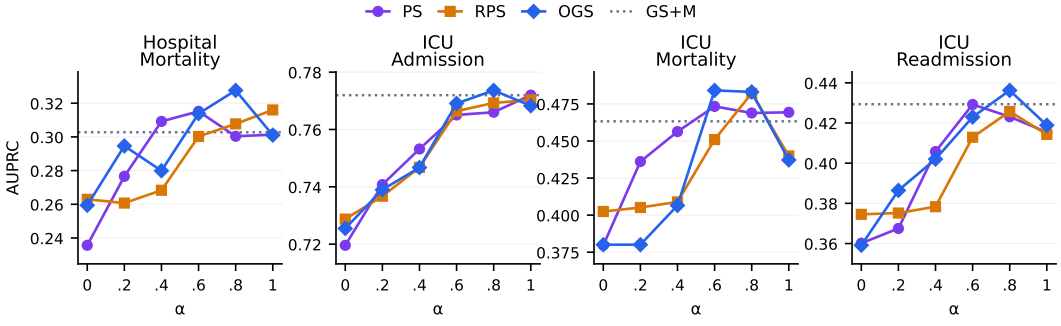
C.1 INITIAL α SWEEP



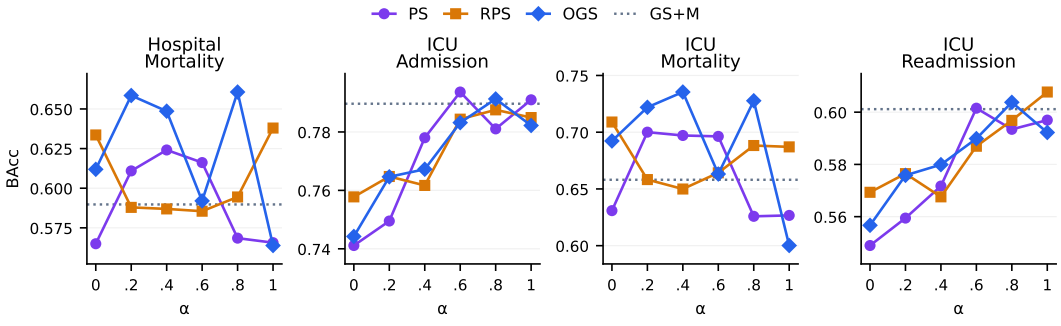
(a) F1 by clinical task.



(b) AUROC by clinical task.



(c) AUPRC by clinical task.



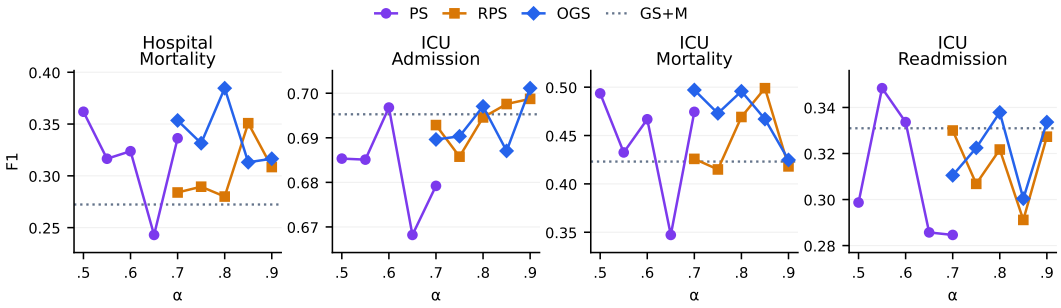
(d) Balanced accuracy by clinical task.

Figure 10: Initial patient-weighting sweep on MIMIC-IV v2.2 with ETHOS tokenization (seed 42). Each column shows one clinical prediction task; rows show F1, AUROC, AUPRC, and balanced accuracy. Curves compare PS, RPS, and OGS across values of the sampling exponent α. Dotted lines show the GS+M baseline, which does not depend on α.

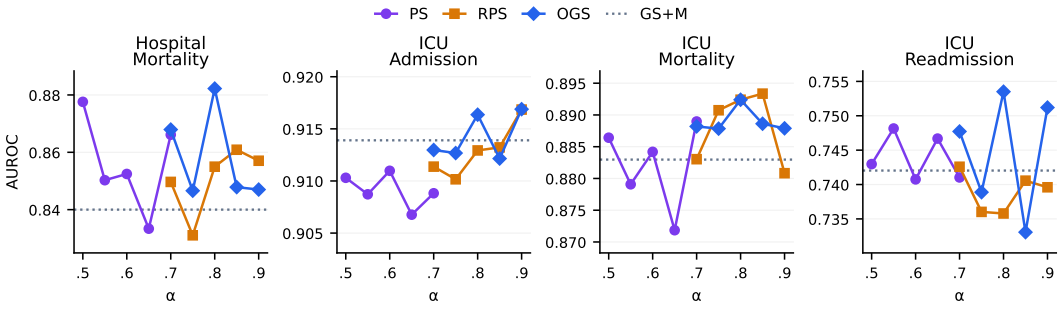
Table 3: Initial single-seed patient-weighting sweep on MIMIC-IV v2.2 (seed 42). Downstream macro metrics average four clinical tasks. Best validation loss is the lowest loss reached during each run. Bold indicates the best value within each method for each metric.

Method	α	Best validation loss ↓	Macro AUROC ↑	Macro AUPRC ↑	Macro F1 ↑	Macro BAcc ↑
PS	0.0	0.9310	0.8095	0.4238	0.3459	0.6215
	0.2	0.9048	0.8307	0.4552	0.4050	0.6550
	0.4	0.8826	0.8463	0.4811	0.4328	0.6677
	0.6	0.8673	**0.8471**	**0.4957**	**0.4553**	**0.6769**
	0.8	0.8588	0.8399	0.4896	0.3992	0.6422
	1.0	**0.8556**	0.8343	0.4895	0.4046	0.6451
RPS	0.0	0.9305	0.8238	0.4422	0.4137	0.6674
	0.2	0.9103	0.8235	0.4444	0.3984	0.6468
	0.4	0.8908	0.8309	0.4506	0.3893	0.6415
	0.6	0.8719	0.8393	0.4826	0.4229	0.6553
	0.8	0.8595	**0.8490**	**0.4962**	0.4414	0.6668
	1.0	**0.8576**	0.8479	0.4851	**0.4598**	**0.6794**
OGS	0.0	0.9311	0.8229	0.4310	0.3919	0.6513
	0.2	0.9098	0.8372	0.4500	0.4299	0.6802
	0.4	0.8905	0.8383	0.4588	0.4386	0.6828
	0.6	0.8722	0.8428	0.4975	0.4281	0.6571
	0.8	0.8609	**0.8611**	**0.5051**	**0.4788**	**0.6959**
	1.0	**0.8582**	0.8339	0.4813	0.3813	0.6346

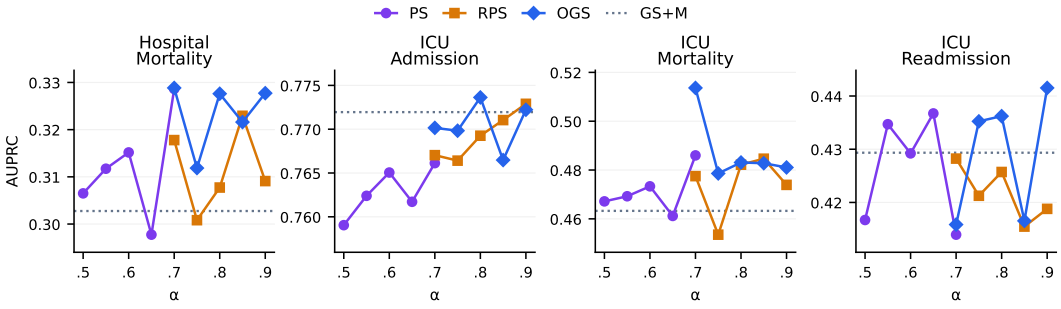
C.2 LOCAL α SWEEP



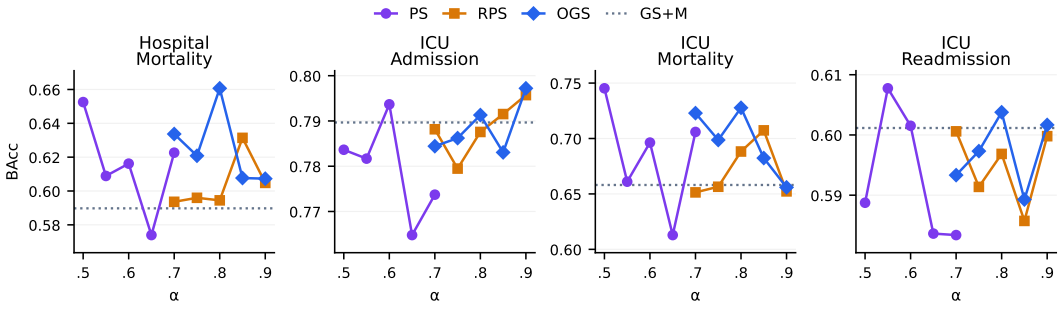
(a) F1 by clinical task.



(b) AUROC by clinical task.



(c) AUPRC by clinical task.



(d) Balanced accuracy by clinical task.

Figure 11: Local patient-weighting sweep on MIMIC-IV v2.2 with ETHOS tokenization (seed 42), following the initial coarse sweep. Each column shows one clinical prediction task; rows show F1, AUROC, AUPRC, and balanced accuracy. Curves compare PS, RPS, and OGS across local values of the sampling exponent α. Dotted lines show the GS+M baseline, which does not depend on α.

Table 4: Local patient-weighting search on MIMIC-IV v2.2 following the initial coarse sweep (seed 42). Downstream macro metrics average four clinical tasks. Best validation loss is the lowest loss reached during each run. Bold indicates the best value within each method for each metric.

Method	α	Best validation loss ↓	Macro AUROC ↑	Macro AUPRC ↑	Macro F1 ↑	Macro BAcc ↑
PS	0.50	0.8733	**0.8543**	0.4874	**0.4599**	**0.6926**
	0.55	0.8703	0.8466	0.4945	0.4456	0.6648
	0.60	0.8673	0.8471	0.4957	0.4553	0.6769
	0.65	0.8641	0.8397	0.4893	0.3860	0.6338
	0.70	**0.8623**	0.8512	**0.4987**	0.4436	0.6714
RPS	0.70	0.8641	0.8467	0.4976	0.4332	0.6584
	0.75	0.8617	0.8420	0.4855	0.4243	0.6558
	0.80	0.8595	0.8490	0.4962	0.4414	0.6668
	0.85	0.8585	**0.8520**	**0.4985**	**0.4596**	**0.6790**
	0.90	**0.8578**	0.8486	0.4937	0.4382	0.6632
OGS	0.70	0.8651	0.8542	**0.5071**	0.4627	0.6836
	0.75	0.8628	0.8465	0.4989	0.4543	0.6757
	0.80	0.8609	**0.8611**	0.5051	**0.4788**	**0.6959**
	0.85	0.8594	0.8454	0.4968	0.4419	0.6656
	0.90	**0.8586**	0.8507	0.5056	0.4440	0.6655

C.3 RESULTS

Table 5: Macro downstream performance across four clinical prediction tasks on MIMIC-IV v2.2, reported as mean ± standard deviation over five seeds. GS denotes Global Stream, GS+M Global Stream with masking, PS Patient Sampling, RPS Recursive Patient Sampling, and OGS Ordered Global Stream. Best mean results are shown in bold and second-best results in italics. Higher values are better.

Method	α	Macro F1	Macro AUROC	Macro AUPRC	Macro BAcc
GS	–	0.420 ± 0.017	0.844 ± 0.004	0.486 ± 0.010	0.655 ± 0.010
GS+M	–	0.430 ± 0.015	0.845 ± 0.004	*0.492 ± 0.004*	0.660 ± 0.009
PS	0.70	0.419 ± 0.032	0.839 ± 0.012	0.481 ± 0.018	0.654 ± 0.018
RPS	0.85	*0.439 ± 0.014*	*0.847 ± 0.005*	0.492 ± 0.007	*0.665 ± 0.010*
OGS	0.80	$\mathbf{0.459 \pm 0.017}$	$\mathbf{0.849 \pm 0.008}$	$\mathbf{0.499 \pm 0.007}$	$\mathbf{0.679 \pm 0.014}$

D TOKENIZATION DETAILS

ETHOS tokenization. For the MIMIC-IV experiments using ETHOS tokenization, we apply the released ETHOS preprocessing and tokenizer to the hospital, ICU, and emergency department records. ETHOS represents clinical events, demographics, measurements, and elapsed time as discrete tokens. It decomposes hierarchical codes according to their coding systems: for example, ICD-10-CM diagnoses are split into a three-character prefix and subsequent components, while ATC medication codes are split into a prefix, fourth character, and suffix. Eligible numeric measurements are assigned to one of ten quantile bins, with boundaries estimated from the training data. Discrete elapsed-time tokens are inserted between events according to the time interval represented by each gap.

Unified data extraction. To test whether the sequence-composition results extend to another EHR representation, we developed dataset-specific DuckDB SQL adapters that convert MIMIC-IV v2.2, eICU-CRD v2.0, and INSPIRE v1.4.2 into chronologically ordered Medical Event Data Standard (MEDS) events. The MIMIC-IV adapter combines hospital and ICU records with linked emergency department records. It includes additional medication administration details, medication reconciliation, microbiology, outpatient measurements, and ICU input, output, and procedure events (Table 6). The eICU-CRD adapter processes admission information, diagnoses, laboratory results, vital signs, medications and infusions, intake and output, treatments, and respiratory charting. The INSPIRE adapter processes operations, diagnoses, laboratory results, ward and perioperative vital signs, and medications. Where a dataset records times as offsets rather than calendar timestamps, its adapter places events on a consistent synthetic timeline while preserving their relative order and intervals.

We retain patients with at least four distinct event timestamps and split each dataset by patient into training, validation, and test sets. For MIMIC-IV, we map ICD-9 diagnoses and procedures to ICD-10 where possible and link medication identifiers to ATC codes using NDC and GSN mappings. Codes without a suitable mapping retain their source identity where supported.

Shared event tokenization. We apply a shared tokenization configuration to the MEDS events from all three datasets, fitting its vocabulary separately on each training split. Each event is serialized into semantic base tokens describing its clinical role and code, followed where applicable by its quantile-binned measurement, unit, or categorical attributes. As in ETHOS, eligible numeric measurements use ten quantile bins. We also use the same elapsed-time tokens and interval rules as ETHOS, inserting them between events at different timestamps. Quantile boundaries and other frequency-dependent tokenization rules are fitted on the training split and applied to the corresponding validation and test splits.

Our representation of hierarchical codes differs from ETHOS. We split a code into its clinical role, coding system, and individual alphanumeric characters. For example, an ICD-10-CM diagnosis of `I25.10` is represented by the base-token sequence `DIAGNOSES`, `ICD-10`, `I`, `2`, `5`, `1`, `0`. Medication orders and administrations retain distinct event roles, with dose and route information included when available.

Event-scoped BPE. For each dataset, we train a Byte-Pair Encoding (BPE) tokenizer with a budget of up to 1,000 merges. Unlike text BPE, which begins with characters or bytes, our procedure begins with the semantic base tokens described above; each base token is treated as an indivisible symbol. A merge is permitted only within a MEDS event or a group of related event components, such as those describing one medication order. Merges therefore cannot combine unrelated clinical events or cross timestamps. Frequent combinations of codes and their attributes can become compact tokens while the boundaries between successive events remain intact.

Table 6: Selected MIMIC-IV source tables included in the ETHOS and unified tokenization configurations. The unified column reflects the repository's current default MIMIC-IV configuration.

MIMIC-IV table	ETHOS	Unified
Demographics and encounters		
`patients`	✓	✓
`admissions`	✓	✓
`transfers`	✓	✓
`triage`	✓	✓
Diagnoses and billing procedures		
`diagnoses_icd`	✓	✓
`procedures_icd`	✓	✓
`drgcodes`	✓	✓
`hcpcsevents`	✓	✓
Medications		
`prescriptions`	✓	✓
`emar`	×	✓
`emar_detail`	×	✓
`pharmacy`	×	×
`medrecon`	×	✓
`pyxis`	×	✓
Laboratory results and vital signs		
`labevents`	✓	✓
`vitalsign`	✓	✓
`chartevents`	✓	×
`microbiologyevents`	×	✓
`omr`	×	✓
ICU events		
`inputevents`	×	✓
`outputevents`	×	✓
`procedureevents`	×	✓

E EVALUATION

Prediction instances. We evaluate on the patient-held-out test split. Each occurrence of a task's prediction-point token can produce a separate prediction instance, so a patient may contribute more than one instance. The prompt includes the prediction point and other events recorded at the same timestamp. We retain the demographic prefix and the most recent clinical history within the model's 2,048-token context window. For tasks without a fixed time horizon, the observed label is determined by the first subsequent terminating outcome. Instances with no forward prediction interval are excluded.

The four MIMIC-IV tasks are evaluated with both tokenizers on v2.2 and with ETHOS tokenization on v3.1. INSPIRE supports hospital readmission, ICU admission, and postoperative ICU admission. eICU-CRD supports hospital mortality, ICU readmission, and short ICU stay. Table 7 gives each task's prediction point and outcome conditions.

Time-limited tasks. Hospital readmission is positive when another hospital admission occurs within 30 days of discharge. Short ICU stay is positive when ICU discharge occurs within 24 hours of admission. If the observed trajectory extends beyond the horizon without the positive outcome, the instance is negative. If it ends before the horizon without a definitive outcome, the instance is excluded because follow-up is incomplete. Observed labels use event timestamps. During generation, elapsed time is estimated from the midpoint durations of the generated ETHOS-style time-interval tokens.

Rollout-based risk estimation. For each prediction instance, we sample 20 continuations at temperature 1. Generation stops when a task-specific outcome token appears, when an applicable time horizon is crossed, or when the maximum number of generated tokens is reached. Crossing a time horizon without the positive outcome counts as a negative rollout. Rollouts that reach the token limit without a valid outcome are excluded from the risk estimate. We estimate risk as the fraction of completed rollouts reaching the positive outcome.

The generation limit is 8,192 tokens for all four MIMIC-IV v2.2 tasks with unified tokenization and all three eICU-CRD tasks, for both GS+M and OGS. All remaining settings use 4,096 tokens. Table 8 summarizes these limits.

Metrics and repetitions. We calculate AUROC and AUPRC from the prediction-instance risk estimates. F1 and balanced accuracy use a risk threshold of 0.5. Dataset-level macro scores are unweighted means across the tasks evaluated in each dataset and tokenization setting. The primary MIMIC-IV v2.2 comparison reports mean and standard deviation across five independently trained model seeds. Each checkpoint undergoes one stochastic rollout evaluation; we do not separately estimate variation from repeating rollouts for a fixed checkpoint. The four additional dataset and tokenization settings use one trained seed per method.

Table 7: Prediction points and outcomes for the rollout benchmarks. The positive outcome must precede the competing negative outcome and, where specified, occur within the time horizon. For horizon-based tasks, TIMELINE_END also stops generated rollouts; observed instances ending before the horizon without a definitive outcome are excluded.

Benchmark	Prediction point	Positive / negative outcome	Horizon
Hospital mortality	HOSPITAL_ADMISSION	MEDS_DEATH (+) HOSPITAL_DISCHARGE (−)	–
ICU mortality	ICU_ADMISSION	MEDS_DEATH (+) ICU_DISCHARGE (−)	–
ICU readmission	ICU_DISCHARGE	ICU_ADMISSION (+) HOSPITAL_DISCHARGE (−)	–
ICU admission	HOSPITAL_ADMISSION	ICU_ADMISSION (+) HOSPITAL_DISCHARGE (−)	–
Hospital readmission	HOSPITAL_DISCHARGE	HOSPITAL_ADMISSION (+) MEDS_DEATH or horizon (−)	30 days
Postoperative ICU admission	OPERATION_END	ICU_ADMISSION (+) HOSPITAL_DISCHARGE (−)	–
Short ICU stay	ICU_ADMISSION	ICU_DISCHARGE (+) MEDS_DEATH or horizon (−)	24 hours

Table 8: Maximum generated tokens per rollout. Methods compared within each dataset and tokenization setting use the same limit.

Dataset	Tokenization	Maximum tokens
MIMIC-IV v2.2	ETHOS	4,096
MIMIC-IV v2.2	Unified	8,192
MIMIC-IV v3.1	ETHOS	4,096
INSPIRE	Unified	4,096
eICU-CRD	Unified	8,192

Table 9: Test-set prediction instances for the indicated dataset and tokenization settings. A patient may contribute multiple instances. Prevalence is the percentage of instances with a positive label.

Task	Instances	Positive	Prevalence
MIMIC-IV v2.2, ETHOS tokenization			
Hospital mortality	42,973	836	1.95%
ICU admission	42,974	6,636	15.44%
ICU mortality	7,365	532	7.22%
ICU readmission	6,469	850	13.14%
MIMIC-IV v2.2, unified tokenization			
Hospital mortality	42,737	782	1.83%
ICU admission	42,737	6,557	15.34%
ICU mortality	7,240	531	7.33%
ICU readmission	6,316	792	12.54%
MIMIC-IV v3.1, ETHOS tokenization			
Hospital mortality	55,528	1,181	2.13%
ICU admission	55,525	8,503	15.31%
ICU mortality	9,551	755	7.90%
ICU readmission	8,249	1,142	13.84%
INSPIRE, unified tokenization			
Hospital readmission (30 days)	10,947	462	4.22%
ICU admission	12,654	1,340	10.59%
Postoperative ICU admission	13,035	1,246	9.56%
eICU-CRD, unified tokenization			
Hospital mortality	15,731	1,455	9.25%
ICU readmission	17,281	2,112	12.22%
Short ICU stay (24 hours)	19,902	7,004	35.19%

F DATASET GENERALISATION

Table 10: Task-level generalization results across datasets and tokenizations for seed 42. GS+M and OGS are compared on each task; BAcc denotes balanced accuracy. Dataset macro scores are unweighted means over the tasks displayed within each dataset. The overall mean weights the five dataset settings equally, and OGS gain is the difference in percentage points, calculated from full-precision macro scores. Bold indicates the better displayed result in each pair; values tied after rounding to three decimal places are both bolded. Task definitions may differ across datasets, so absolute scores should not be compared directly between them.

	AUROC		AUPRC		F1		BAcc	
Task	GS+M	OGS	GS+M	OGS	GS+M	OGS	GS+M	OGS
MIMIC-IV v2.2								*ETHOS tokenization*
Hospital mortality	0.836	**0.868**	0.310	**0.329**	0.265	**0.353**	0.585	**0.634**
ICU admission	0.911	**0.913**	0.769	**0.770**	**0.690**	**0.690**	**0.784**	**0.784**
ICU mortality	0.886	**0.888**	0.462	**0.514**	0.431	**0.497**	0.662	**0.723**
ICU readmission	0.735	**0.748**	**0.420**	0.416	**0.312**	0.311	**0.593**	**0.593**
Macro	0.842	**0.854**	0.491	**0.507**	0.424	**0.463**	0.656	**0.684**
MIMIC-IV v2.2								*Our tokenization*
Hospital mortality	0.739	**0.831**	0.217	**0.267**	0.075	**0.185**	0.520	**0.555**
ICU admission	0.914	**0.916**	0.765	**0.766**	0.678	**0.681**	0.773	**0.776**
ICU mortality	0.773	**0.833**	0.313	**0.361**	0.176	**0.330**	0.549	**0.616**
ICU readmission	0.725	**0.751**	0.387	**0.403**	**0.256**	0.243	**0.573**	0.569
Macro	0.788	**0.833**	0.420	**0.449**	0.296	**0.360**	0.604	**0.629**
MIMIC-IV v3.1								*ETHOS tokenization*
Hospital mortality	**0.855**	0.851	0.335	**0.349**	0.342	**0.385**	0.625	**0.651**
ICU admission	0.910	**0.912**	**0.764**	**0.764**	0.678	**0.681**	0.777	**0.781**
ICU mortality	0.865	**0.867**	**0.476**	0.469	0.450	**0.463**	0.673	**0.691**
ICU readmission	0.711	**0.726**	0.401	**0.410**	0.275	**0.296**	0.580	**0.587**
Macro	0.835	**0.839**	0.494	**0.498**	0.436	**0.456**	0.663	**0.677**
INSPIRE								*Our tokenization*
Hospital readmission	0.789	**0.793**	0.300	**0.310**	0.224	**0.240**	0.568	**0.573**
ICU admission	0.912	**0.920**	0.655	**0.659**	0.543	**0.599**	0.700	**0.741**
Postoperative ICU admission	0.957	**0.960**	**0.831**	0.820	0.755	**0.764**	0.834	**0.858**
Macro	0.886	**0.891**	0.595	**0.596**	0.507	**0.534**	0.700	**0.724**
eICU-CRD								*Our tokenization*
Hospital mortality	0.649	**0.688**	0.205	**0.230**	0.098	**0.132**	0.525	**0.534**
ICU readmission	**0.779**	0.762	**0.436**	0.430	**0.411**	0.343	**0.644**	0.607
Short ICU stay	0.706	**0.715**	0.586	**0.597**	0.384	**0.465**	0.604	**0.630**
Macro	0.711	**0.722**	0.409	**0.419**	0.298	**0.313**	**0.591**	0.590
Across dataset settings								
Mean of dataset macros	0.813	**0.828**	0.482	**0.494**	0.392	**0.425**	0.643	**0.661**
OGS gain	+1.52		+1.22		+3.29		+1.80	