One Loss to Rule Them All: Marked Time-to-Event for Structured EHR Foundation Models

Zilin Jing [1] **Vincent Jeanselme** [1] **Yuta Kobayashi** [1] **Simon A. Lee** [2] **Chao Pang** [1 3] **Aparajita Kashyap** [1] **Yanwei Li** [1] **Xinzhuo Jiang** [1] **Shalmali Joshi** [1]

Abstract

Clinical events captured in Electronic Health Records (EHR) are irregularly sampled and may consist of a mixture of discrete events and numerical measurements, such as laboratory values or treatment dosages. The sequential nature of EHR, analogous to natural language, has motivated the use of next-token prediction to train prior EHR Foundation Models (FMs) over events. However, this training fails to capture the full structure of EHR. When a given event occurs must be captured, but the event value (abnormal lab) also modulates the likelihood of other clinical events. Most existing EHR FMs do not jointly model this likelihood and are unable to capture the full observation process, impacting downstream capabilities. We propose *ORA*, a marked time-to-event pretraining objective that jointly models event timing and associated measurements. Across multiple datasets, downstream tasks, and model backbones, this objective consistently yields more generalizable representations than next-token prediction and pretraining losses that ignore continuous measurements. Importantly, the proposed objective yields improvements beyond traditional classification evaluation, including better regression and time-to-event prediction. Beyond introducing a new family of FMs, our ablations suggest a broader takeaway: pretraining objectives that account for EHR structure are critical for expanding downstream capabilities and generalizability.

1. Introduction

Large-scale FMs have shifted the paradigm of machine learning in health by extracting representations that transfer across tasks, cohorts, and institutions (Guo et al., 2023; 2024; Lee et al., 2025; Burkhart et al., 2025). Inspired by the success of large language models (LLMs), recent work has framed Electronic Health Records (EHR) as an autoregressive sequence of discrete events pretrained using next-token prediction analogous to those used in natural language processing. These approaches have demonstrated state-of-the-art performance on a range of downstream classification tasks (Li et al., 2020; Pang et al., 2024; 2021b; Odgaard et al., 2024).

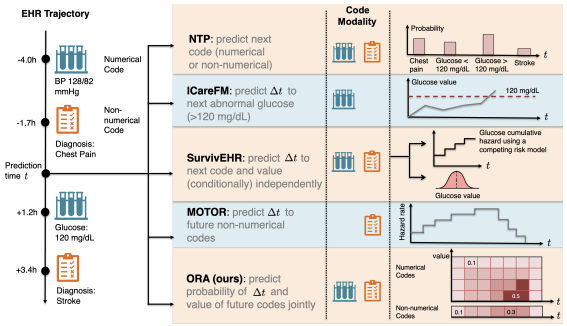

Figure 1. **ORA: Joint Modeling of Event Time and Value in EHR Pretraining.**Comparison of conventional EHR pretraining objectives with ORA, illustrating how prior methods separately or partially model event sequences, whereas ORA jointly captures the distribution over future event timing and associated numerical values within irregular, heterogeneous clinical trajectories.

Despite these advances, EHR data fundamentally differ from natural language in ways that challenge this analogy, motivating subsequent work. Clinical records are not sequences of tokens. Instead, they consist of irregularly sampled heterogeneous medical events, often accompanied by numerical measurements. To account for these deviations, most FMs design customized tokenizations, including artificial time tokens and time embeddings (Pang et al., 2024) for irregularity, discretized intervals (Wornow et al., 2024), and digit space embedding (Hur et al., 2023) for numerical value. However, these modifications focus on how data should be represented as input and often overlook the design of EHR pretraining losses.

Recent literature (Steinberg et al., 2024; Gadd et al., 2025; Burger et al., 2025) proposed time-to-event pretraining loss as an alternative, with Gadd et al. (2025); Burger et al. (2025) additionally introducing pretraining objectives that also capture numerical values. However, these works often

[1]Columbia University [2]UCLA [3]Formation Bio. Correspondence to: Zilin Jing <zj2398@columbia.edu>.

Proceedings of the Workshop on Structured Data for Health at the 43rd International Conference on Machine Learning, Seoul, South Korea. Copyright 2026 by the author(s).

introduce novel architectural choices, preprocessing, and tokenization strategies in addition to new pretraining losses. As a result, the impact of each design choice has been challenging to isolate. Further, when prior works model time-to-event (TTE) and mixed-event types, they either assume conditional independence between the time and value or model only abnormal values. A key missing element is the ability to jointly model time-to-event and mixed-event types (i.e., model values, if available, and marginalize them if not). We motivate this as a feasible tradeoff to a full joint likelihood of time, event-type, and values.

We propose ORA, a marked time-to-event pretraining objective that jointly models the distribution of event timing and associated measurements for each medical code, naturally handling irregular intervals, censoring, and associated numerical values, if present. For the first time, we expand downstream evaluation of EHR FMs to characterize time-to-event and regression performance on multiple clinically meaningful tasks (Pang et al., 2025), to expand the clinical utility of EHR FMs. Across two large EHR datasets, representing large tertiary and quaternary healthcare institutions, we extensively evaluate the pretraining objective across 15 tasks. We contribute a family of FMs, demonstrating that our proposed loss can be integrated with Transformer and Mamba backbones with improved performance on average across 7 classification, 4 time-to-event, and 4 regression tasks across two different EHR datasets. Together, these results provide a critical insight into the design of future FMs: pretraining objectives should align with the full structure of EHR data to improve expressiveness, generalizability, and downstream capability. Our contributions can be summarized as follows:

- **Novel pre-training loss.** Building on the parallel between EHR and marked point processes, we introduce ORA, a composite code-specific likelihood to capture the complexity of EHR models.

- **Comprehensive evaluation and ablations.** Our evaluation is comprehensive: 7 binary classification, 4 TTE prediction, and 4 regression tasks across two datasets (MIMIC-IV (Johnson et al., 2023) and Columbia University Irving Medical Center (CUMC)). Our ablations demonstrate the crucial impact of the choice of pre-training loss for structured EHR FMs.

- **Generalizability.** We demonstrate improved generalizability of ORA using both Mamba and Transformer base models (Section 4), across all tasks and dataets.

2. Formulation

Consider each patient $i \in [1, 2, \cdots, N]$'s EHR data to consist of a tuple $\mathcal{H}_i := \{(t, m, v)_{i,j}, j \in [1, 2, \cdots, N_i]\}$ where $t \in \mathbb{R}^+$, is the timestamp of the event, $m \in \mathcal{M}$ the clinical code associated with the event, such as ICD-10 diagnosis or procedure codes, and $v \in \mathbb{R} \cup \{\varnothing\}$ the optional numerical value associated with the event. N_i denotes the number of events observed for patient i. For a patient i, events up to j (ordered by time) are denoted by $\mathcal{H}_{i,<j}$. That is, $\mathcal{H}_{i,<j} := \{(t, m, v)_{i,l}, l \in [1, 2, \cdots, j-1]\}$. An intuitive approach is to maximize the joint likelihood of the full marked point process, corresponding to the joint between the time to the next event, the associated code, and value. Under standard IID assumptions on patient samples, the joint likelihood can be decomposed as:

$$\mathcal{L}(\theta) = \prod_{i=1}^{N} p_\theta(\mathcal{H}_i) = \prod_{i=1}^{N} \prod_{j \in [1..N_i]} p_\theta((t, m, v)_{i,j} \mid \mathcal{H}_{i,<j}) \tag{1}$$

Modeling the complete joint likelihood in Equation (1) is challenging since we need to capture the complex relationship between time, code, and value. EHR events also co-occur frequently and happens in irregularly periods. We comapre different pretrained loss functions in Appendix C) and analyse how they fails to accurately model the true joint likelihood.

3. Method: ORA

To address these challenges, we instead propose optimizing the composite code-specific likelihood. We denote by $f_{i,j}^m$, the first occurrence of code m after the time associated with event j, and the set of the first events of all codes as $\mathcal{F}_{i,j} = \{\forall m, f_{i,j}^m\}$. Specifically, we define:

$$f_{i,j}^m = (\Delta t_j^m, v, \delta, m)$$

where, if an event of type m occurs after t_j, Δt_j^m corresponds to the interval between the current time t_j and the time to the next event m, v is the associated numerical value with the future event, and $\delta = 1$ as an event indicator. If no such event occurs in the future, we denote $\Delta t_j^m = t_{i,N_i} - t_{i,j}$ as the interval up to the last observation, and set the event indicator to $\delta = 0$. In this context, δ denotes whether an event m is observed within the recorded EHR for patient i after the observation j or censored. Accounting for such unobserved events is critical, as ignoring them leads to biased likelihood estimates (Chen et al., 2024). Suppose all the first occurrences of future codes are conditionally independent given the contextual history $\mathcal{H}_{i,<j}$ of the patient: $\mathcal{L}_{\text{ORA}}(\theta) = \prod_i \prod_j \prod_{(\Delta t, m, v, \delta) \in \mathcal{F}_{i,j}} p_{\theta_m}(\Delta t, v, \delta \mid \mathcal{H}_{i,<j})$.

Formally, this corresponds to a code-specific marked point process. This formulation predicts the joint probability of time and value, capturing the complex dependence between different properties within a medical event. Meanwhile,

it empirically allows the use of all potential next events at a given time point, thereby overcoming the sparsity of observations and improving data efficiency. Finally, ORA is robust for both nonnumerical and numerical events. If v is missing or empty, we can marginalize the probability over v. The detailed comparison between ORA and existing EHR FMs losses can be found in Table D.

3.1. Discretized ORA

The central challenge in computing the proposed ORA loss is the estimation of $p_{\theta_m}(\Delta t, v, \delta \mid \mathcal{H}_{i,<j})$, or equivalently, under the conditional independence and the common uninformative censoring assumptions: $\log p_{\theta_m}(\Delta t, v, \delta \mid \mathcal{H}_{<j}) = \delta \log \lambda_{\theta_m}(t, v \mid \mathcal{H}_{<j}) - \int_0^t \int_{v'} \lambda_{\theta_m}(u, v' \mid \mathcal{H}_{<j}) dv' dt'$, where $\lambda_{\theta_m}(\cdot)$ is the instantaneous risk, or the intensity function, of observing an event code m.

Estimating the likelihood, therefore, requires modeling the instantaneous risk and its integral for each code m. In this work, we adopt a discretization of the joint, extending survival approaches such as DeepHit (Lee et al., 2018) to marked point processes. While approximating the likelihood, this approach has been shown to be computationally efficient and to achieve state-of-the-art performance in survival analysis (Lee et al., 2018). Concretely, we add a final network layer to model the discretized joint distribution over time-value tuples. For each code m and input x, the neural network outputs a matrix $P^m(x) \in [0,1]^{T \times V}$, a discretized approximation of the intensity function, where T and V denote the number of discretized bins for time and value, respectively. Each entry $P^m[k,l]$ represents the probability of observing an event in the kth time-quantile and l^{th} value-quantile. Note that these quantiles are code-specific to ensure more flexible modeling. Let $q_m(t,v)$ return the time-value quantiles associated with code m to which (t,v) belongs. When v is omitted, $q_m(t,\cdot)$ denotes the time-quantile. Under this discretization, the log-likelihood can be expressed as:

$$\log \tilde{\mathcal{L}}_{\text{ORA}}(\theta) := \sum_{i=1}^{N} \sum_{j \in [1 \ldots N_i]} \sum_{(t,m,v,\delta) \in \mathcal{F}_{i,j}} (1-\delta) \log \Big(1 - \sum_{k=1}^{q_m(t,\cdot)} \sum_{l} P_\theta^m[k,l](\mathcal{H}_{i,<j}) + \delta \log P_\theta^m[q_m(t,v)](\mathcal{H}_{i,<j})$$

where $q_m((t,v))$ returns the quantile to which (t,v) belongs for mark/event m. More interpretations of this likelihood can be found in Appendix E.4.

3.2. Model design

To maximize the discretized ORA likelihood, one must extract a representation of the medical history $\mathcal{H}_{i,<j}$ and compute the matrix $P^m(x)$. Importantly, the proposed pre-training loss is backbone agnostic. We propose evaluating its efficacy on two common FMs backbones: the attention-based Transformer (Vaswani et al., 2017), and a state-space model (Mamba (Gu & Dao, 2024)) to demonstrate its utility. The detaild tokenization and backbone set up can be found in Appendix E.3.

4. Experiments

Our experiments characterize the impact of the choice of pretraining loss on the representativeness of extracted embeddings. We compare prior EHR FMs to ORA using two base backbones and conduct linear probing to evaluate performance on 15 tasks across two EHR datasets. Our ablations further explicitly compare NTP, TPP, and ORA to isolate the impact of the proposed pre-training loss.

4.1. Datasets

We rely on one public EHR dataset: MIMIC-IV (Johnson et al., 2023) and a private EHR dataset from Columbia University Irving Medical Center (CUMC). MIMIC-IV contains 364,000 patients who were admitted to Beth Israel Deaconess Medical Center between 2008 and 2022, with tabular time-series measurements of diagnoses, medications, lab tests, etc. The CUMC dataset contains 6.7 million patients with both inpatient and outpatient visits, primary care, and emergency services between 1986 and 2023, consisting of diagnoses, procedures, medications, and visit information (see Appendix E.1 for patient split used for pretraining).

4.2. Downstream tasks

To ensure clinical utility, we design a set of clinically meaningful downstream tasks that include 7 binary classification tasks, 4 time-to-event prediction tasks and 4 regression tasks. The detailed definitions can be found in Appendix E.2.

4.3. Baseline models and associated loss

Our experiments include the following baselines and FMs:

- **Task-Specific Baselines.** For each task, we consider XGBoost (Chen & Guestrin, 2016) for binary classification, Deephit (Lee et al., 2018) for time-to-event prediction based on FEMR features (Steinberg et al., 2024) from the EHR time series. For regression, we used the most recent value for the same lab code measured before the prediction time, which is a strong baseline that outperforms both structured EHR foundation models and general LLM (GPT-4o) in lab outcome predictions (Im et al., 2025).
- **Context Clue** (Wornow et al., 2024) is a next-token prediction (NTP) EHR FM family. We include two different backbones: CC-Llama and CC-Mamba.

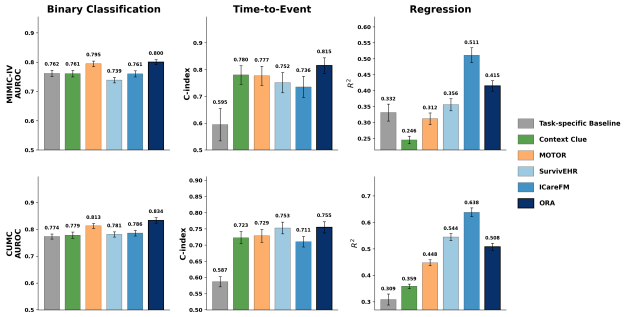

Figure 2. **ORA Improves Generalization Across Classification, Time-to-Event, and Regression Tasks.** Average performance on MIMIC-IV and Columbia University Irving Medical Center (CUMC) datasets shows that jointly modeling event timing and values yields consistently stronger representations. ORA outperforms different foundation models across both binary classification and TTE tasks and remains competitive in regression results. All models used the **Transformer** backbone. Full results for each task can be found in Appendix G.

- **Motor** (Steinberg et al., 2024) is a time-to-event (TTE) EHR foundation model pretrained to predict code-specific future event risks. It serves as an external baseline for temporal point process-style pretraining, which models event timing but does not explicitly model the associated numerical value distribution.

- **SurvivEHR** (Gadd et al., 2025) and **ICareFM** (Burger et al., 2025) are two marked time-to-event foundation models (MTPPs) that jointly predict time and value in the pretraining losses. These baselines are closest in spirit to ORA, while differing in their loss functions.

4.4. Ablation studies

To exclusively study the impact of loss function and understand how each component (medical code, time and numerical value) in our ORA loss contributes to the downstream performance, we fix the tokenizer and backbone of ORA. We create three ablations with the loss functions as follows:

- **NTP.** We train standard next-token prediction baselines that autoregressively predict the code type of the next medical event using cross-entropy loss. This objective supervises the next code only, without directly modeling event time or numerical value.

- **TPP.** We train temporal point process-style baselines that predict, for each medical code, the time interval to its first future occurrence. This objective adds time-to-event supervision while marginalizing over the numerical values.

- **ORA.** We train the full marked time-to-event objective proposed in this work. In addition to predicting the time interval as in TPP, ORA jointly models the value distribution for numerical codes such as laboratory tests and vital signs.

4.5. Evaluation

To quantify the discriminative performance of all models, we report the Area Under the Receiver Operating Characteristic Curve (AUROC) for classification, the time-dependent C-index (Antolini et al., 2005) for TTE, and R^2 coefficient for regression. When evaluating the pretrained models, we generate feature embeddings for each task cohort and fit a task-specific head (logistic regression for binary classification, Deephit for time-to-event tasks, and linear regression for regression tasks).

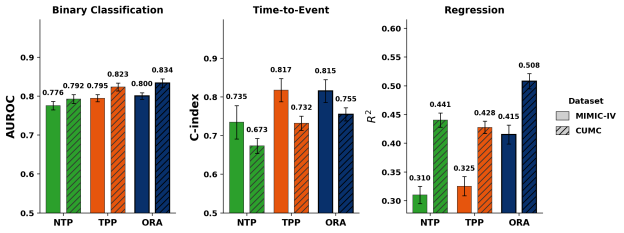

Figure 3. **Pretraining Objective Drives Downstream Performance Across Task Families.** Under a fixed tokenizer and Transformer backbone, jointly modeling time-to-event and values yields consistent gains over NTP and TPP objectives in most tasks, demonstrating that improvements arise from the loss formulation rather than model capacity or input representation.

5. Results

5.1. ORA improves overall task performance across datasets

Figure 2 compares ORA with task-specific baselines and representative EHR foundation models on top of the Transformer backbone. Across both MIMIC-IV and CUMC, ORA achieves the most balanced performance across binary classification, time-to-event prediction, and regression. On MIMIC-IV, ORA obtains the best average AUROC for binary classification and the best average C-index for time-to-event prediction, reaching 0.800 and 0.815, respectively. These results improve over the strongest non-ORA baselines, including MOTOR and Context Clue. On CUMC, the same trend holds: ORA achieves the best average binary classification performance (0.834 AUROC) and remains best or nearly tied for time-to-event prediction (0.755 C-index).

For regression tasks, ORA consistently outperforms NTP and TPP baselines. While specialized MTPP models such as SurvivEHR and ICareFM achieve the highest R^2 in regression tasks, this performance comes at the cost of severe degradation in classification and TTE. This trade-off is expected since ICareFM relies on a surrogate loss function that isolates abnormal numerical events, over-indexing on numeric data while ignoring non-numerical codes. SurvivEHR is sensitive to highly concurrent events, and also assumes the conditional independence between time and value. In

contrast, ORA jointly models the distribution of event timing and associated measurements, achieving the best overall performance across three different types of tasks.

5.2. Ablation studies: isolating the impact of the ORA loss function

To isolate the impact of the pretraining objective, Figure 3 compares different objectives under a fixed tokenizer and Transformer backbone. This ablation directly tests whether the gains of ORA come from its marked time-to-event objective rather than from architectural or tokenization differences. In this controlled setting, ORA consistently outperforms NTP across all three task families on both datasets. These gains show that predicting only the next discrete event is insufficient for learning general-purpose EHR representations, since it ignores both irregular timing and event-associated numerical values.

Compared with TPP, which adds time-to-event supervision but does not jointly model numerical values, ORA preserves the temporal benefits of TTE modeling while substantially improving tasks that depend on measurement values. This comparison shows that time modeling alone is not sufficient: the additional value component in ORA is critical for regression and also improves classification without sacrificing time-to-event performance. Overall, the ablation demonstrates that the marked time-to-event loss itself is the key driver of the observed robustness across tasks and datasets.

5.3. ORA's performance generalizes across backbones

Finally, we demonstrate that the representational gains of the ORA loss are fundamentally backbone-agnostic. We replaced the attention-based Transformer with Mamba, a state-space architecture. Figure 6 in Appendix F further supports this conclusion. Although MOTOR is slightly stronger for binary classification and ICareFM is the strongest for regression, ORA again achieves the most balanced performance across task families, with strong classification, the best time-to-event performance, and competitive regression. Meanwhile, Figure 7 shows that moving from NTP or TPP to ORA loss is also beneficial for Mamba backbones. Together with the Transformer results, these findings suggest that the advantage of ORA comes from the pretraining objective itself rather than a particular backbone. Across datasets and backbones, jointly modeling when clinical events occur and what values they take yields more general-purpose EHR representations than objectives that model only tokens, only time, or only numerical deterioration events.

6. Discussion

We introduce ORA, a marked time-to-event pretraining objective for structured EHR foundation models. The central motivation of this work is that EHR trajectories are not simply sequences of discrete tokens: clinical events occur irregularly over time, may be observed concurrently, and are often associated with numerical values. Existing pretraining objectives only partially capture this structure. Next-token prediction treats EHR as an ordered sequence of discrete codes. Time-to-event objectives model event timing but often ignores the complex dependence between time and value, fails to capture mixed-type events and it is sensitive to concurrent events.

Our empirical results support the importance of aligning the pretraining objective with the structure of EHR data. Across two EHR datasets, three downstream task families, and two backbone architectures, ORA produces representations that are consistently more generalizable than next-token prediction and time-to-event objectives that ignore numerical measurements. In particular, ORA improves not only binary classification, the dominant evaluation setting in prior EHR foundation model work, but also time-to-event prediction and numerical regression. This broader evaluation highlights that EHR foundation models should be assessed beyond classification, since clinically useful representations must support heterogeneous prediction targets.

A key strength of our study is that we isolate the effect of the pretraining loss. By fixing the tokenizer and backbone in our ablation studies, we directly compare ORA against next-token prediction and time-to-event objectives under controlled settings. These experiments show that jointly predicting event timing and value improves downstream performance overall, with especially strong gains on regression tasks. The same trend holds across both Transformer and Mamba backbones, suggesting that the benefit of ORA is not tied to a specific sequence model. More broadly, our results indicate that pretraining objectives, rather than backbone alone, are a critical design choice for structured EHR foundation models.

Limitations and future work. There are several limitations. First, ORA relies on quantile-based discretization of time and value. This choice is simple and scalable, and adapts to code-specific distributions, but it may be suboptimal for sharply peaked or highly multimodal distributions. More expressive discretization schemes, adaptive binning, or continuous likelihood parameterizations could further improve the fidelity of the learned event-time and value distributions.

Second, our composite likelihood assumes that the first future occurrences of different codes are conditionally independent given the patient history. This assumption improves scalability and allows dense supervision over many possible future events, but it is still a simplification of the full joint EHR process. Future work could relax this assumption by modeling dependencies among future codes, while preserving the computational advantages of the current formulation.

References

Al-Sahab, B., Leviton, A., Loddenkemper, T., Paneth, N., and Zhang, B. Biases in electronic health records data for generating real-world evidence: an overview. *Journal of healthcare informatics research*, 8(1):121–139, 2024.

Antolini, L., Boracchi, P., and Biganzoli, E. A time-dependent discrimination index for survival data. *Statistics in medicine*, 24(24), 2005.

Burger, M., Chopard, D., Londschien, M., Sergeev, F., Yèche, H., Kuznetsova, R., Faltys, M., Gerdes, E., Leshetkina, P., Bühlmann, P., et al. A foundation model for intensive care: Unlocking generalization across tasks and domains at scale. *medRxiv*, pp. 2025–07, 2025.

Burkhart, M. C., Ramadan, B., Liao, Z., Chhikara, K., Rojas, J. C., Parker, W. F., and Beaulieu-Jones, B. K. Foundation models for electronic health records: representation dynamics and transferability. *arXiv preprint arXiv:2504.10422*, 2025.

Chen, G. H. et al. An introduction to deep survival analysis models for predicting time-to-event outcomes. *Foundations and Trends® in Machine Learning*, 17(6):921–1100, 2024.

Chen, T. and Guestrin, C. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pp. 785–794. ACM, August 2016.

Fallahpour, A., Alinoori, M., Ye, W., Cao, X., Afkanpour, A., and Krishnan, A. Ehrmamba: Towards generalizable and scalable foundation models for electronic health records. *arXiv preprint arXiv:2405.14567*, 2024.

Gadd, C., Gokhale, K., Acharya, A., Cooper, J., Crowe, F., Fitzsimmons, L., Jackson, T., Nirantharakumar, K., Yau, C., and collaborative, O. Survivehr: a competing risks, time-to-event foundation model for multiple long-term conditions from primary care electronic health records. *medRxiv*, pp. 2025–08, 2025.

Gu, A. and Dao, T. Mamba: Linear-time sequence modeling with selective state spaces. In *First conference on language modeling*, 2024.

Guo, L. L., Steinberg, E., Fleming, S. L., Posada, J., Lemmon, J., Pfohl, S. R., Shah, N., Fries, J., and Sung, L. Ehr foundation models improve robustness in the presence of temporal distribution shift. *Scientific Reports*, 13(1): 3767, 2023.

Guo, L. L., Fries, J., Steinberg, E., Fleming, S. L., Morse, K., Aftandilian, C., Posada, J., Shah, N., and Sung, L. A multi-center study on the adaptability of a shared foundation model for electronic health records. *NPJ digital medicine*, 7(1):171, 2024.

Guo, L. L., Arciniegas, S. E., Lee, J. J., Yan, A. P., Tomlinson, G., Fries, J., and Sung, L. Tokenization tradeoffs in structured ehr foundation models. *arXiv preprint arXiv:2603.15644*, 2026.

Hegselmann, S., von Arnim, G., Rheude, T., Kronenberg, N., Sontag, D., Hindricks, G., Eils, R., and Wild, B. Large language models are powerful electronic health record encoders. *arXiv preprint arXiv:2502.17403*, 2025.

Hur, K., Oh, J., Kim, J., Kim, J., Lee, M. J., Cho, E., Moon, S.-E., Kim, Y.-H., Atallah, L., and Choi, E. Genhpf: General healthcare predictive framework for multi-task multi-source learning. *IEEE Journal of Biomedical and Health Informatics*, 28(1):502–513, 2023.

Im, S., Oh, J., and Choi, E. Labtop: A unified model for lab test outcome prediction on electronic health records. *arXiv preprint arXiv:2502.14259*, 2025.

Johnson, A. E., Bulgarelli, L., Shen, L., Gayles, A., Shammout, A., Horng, S., Pollard, T. J., Hao, S., Moody, B., Gow, B., et al. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1, 2023.

Lee, C., Zame, W., Yoon, J., and Van Der Schaar, M. Deephit: A deep learning approach to survival analysis with competing risks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Lee, S. A., Jain, S., Chen, A., Ono, K., Biswas, A., Rudas, Á., Fang, J., and Chiang, J. N. Clinical decision support using pseudo-notes from multiple streams of ehr data. *npj Digital Medicine*, 8(1):394, 2025.

Li, Y., Rao, S., Solares, J. R. A., Hassaine, A., Ramakrishnan, R., Canoy, D., Zhu, Y., Rahimi, K., and Salimi-Khorshidi, G. Behrt: transformer for electronic health records. *Scientific reports*, 10(1):7155, 2020.

Odgaard, M., Klein, K. V., Thysen, S. M., Jimenez-Solem, E., Sillesen, M., and Nielsen, M. Core-behrt: A carefully optimized and rigorously evaluated behrt. *arXiv preprint arXiv:2404.15201*, 2024.

Pang, C., Jiang, X., Kalluri, K. S., Spotnitz, M., Chen, R., Perotte, A., and Natarajan, K. Cehr-bert: Incorporating temporal information from structured ehr data to improve prediction tasks. In *Machine learning for health*, pp. 239–260. PMLR, 2021a.

Pang, C., Jiang, X., Kalluri, K. S., Spotnitz, M., Chen, R., Perotte, A., and Natarajan, K. Cehr-bert: Incorporating temporal information from structured ehr data to

improve prediction tasks. In Roy, S., Pfohl, S., Rocheteau, E., Tadesse, G. A., Oala, L., Falck, F., Zhou, Y., Shen, L., Zamzmi, G., Mugambi, P., Zirikly, A., McDermott, M. B. A., and Alsentzer, E. (eds.), *Proceedings of Machine Learning for Health*, volume 158 of *Proceedings of Machine Learning Research*, pp. 239–260. PMLR, 04 Dec 2021b. URL https://proceedings.mlr.press/v158/pang21a.html.

Pang, C., Jiang, X., Pavinkurve, N. P., Kalluri, K. S., Minto, E. L., Patterson, J., Zhang, L., Hripcsak, G., Gürsoy, G., Elhadad, N., et al. Cehr-gpt: Generating electronic health records with chronological patient timelines. *arXiv preprint arXiv:2402.04400*, 2024.

Pang, C., Jeanselme, V., Choi, Y. S., Jiang, X., Jing, Z., Kashyap, A., Kobayashi, Y., Li, Y., Pollet, F., Natarajan, K., et al. Fomoh: A clinically meaningful foundation model evaluation for structured electronic health records. *arXiv preprint arXiv:2505.16941*, 2025.

Renc, P., Jia, Y., Samir, A. E., Was, J., Li, Q., Bates, D. W., and Sitek, A. Zero shot health trajectory prediction using transformer. *NPJ Digital Medicine*, 7(1):256, 2024.

Shmatko, A., Jung, A. W., Gaurav, K., Brunak, S., Mortensen, L. H., Birney, E., Fitzgerald, T., and Gerstung, M. Learning the natural history of human disease with generative transformers. *Nature*, 647(8088):248–256, 2025.

Steinberg, E., Xu, Y., Fries, J. A., and Shah, N. MOTOR: A time-to-event foundation model for structured medical records. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=NialiwI2V6.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Wornow, M., Bedi, S., Hernandez, M. A. F., Steinberg, E., Fries, J. A., Ré, C., Koyejo, S., and Shah, N. H. Context clues: Evaluating long context models for clinical prediction tasks on ehrs. *arXiv preprint arXiv:2412.16178*, 2024.

A. Acknowledgments

VJ and SJ would like to acknowledge partial support from NIH 5R01MH137679-02. AK would like to acknowledge the support of NIH 5T15LM007079-34. ZJ and SJ would like to acknowledge partial support from NIH 5R01LM014380-02. SJ and ZJ would like to acknowledge partial support from the Google Research Scholar Award. SAL would like to acknowledge the generous support of the Warren Alpert Foundation, Warren Alpert-UCLA Computational Biology/AI (CBAI) Scholar Training and Retention Program (award 20242540). SJ would like to acknowledge partial support from the SNF Center for Precision Psychiatry & Mental Health, Columbia. Any opinions, findings, conclusions, or recommendations in this manuscript are those of the authors and do not reflect the views, policies, endorsements, expressed or implied, of any aforementioned funding agencies/institutions.

B. Impact Statement

Foundation models in healthcare promise to democratize access to advanced modeling capabilities for institutions lacking the resources to build models from scratch. Yet this promise comes with risks to privacy, fairness, and safety. Although this paper advances the mathematical foundations of these models, such risks must be rigorously addressed before any clinical implementation.

C. Literature Review

We discuss different types of pretrained loss functions used in previous EHR foundation models.

Next Token Prediction (NTP): Inherited from natural language, the NTP loss collects all medical codes m as a vocabulary and predicts the next token in the trajectory autoregressively. This is equal to maximizing a simplified likelihood:

$$\mathcal{L}_{\text{NTP}}(\theta) := \prod_{i=1}^{N} \prod_{j \in [1..N_i]} p_\theta(m_{i,j} \mid \mathcal{H}_{i,<j})$$

However, such losses do not account for the temporal irregularities and associated values characteristic of EHR, a problem that has largely been overlooked in the development of FMs. To overcome these limitations, several NTP models design EHR-specific tokenization strategies: (i) train separate embedding layers for time and value, synthesized with code embeddings as input (Fallahpour et al., 2024; Pang et al., 2021a). Such tokenizations do not treat elapsed time and value as an explicit supervised target, making it hard for embeddings to capture an irregular data-generating process, (ii) adding discretized tokens for time and value to the vocabulary (Hegselmann et al., 2025; Pang et al., 2024; Renc et al., 2024). The cross-entropy loss is computed for all t, m, and v tokens. For example, Context Clue (Wornow et al., 2024) discretizes each numerical measurement to ten intervals and assigns each interval a unique token during pretraining. However, this method blows up the vocabulary size in order to get a precise division of each numerical code. Meanwhile, it ignores censoring in the loss function, which causes structural missingness and incomplete EHR trajectories (Al-Sahab et al., 2024).

Temporal Point Process (TPP): To capture the underlying irregular data-generating process, TPP directly predicts the time interval to the next event and its code type (Shmatko et al., 2025):

$$\mathcal{L}_{\text{TPP}}(\theta) := \prod_{i=1}^{N} \prod_{j} p_\theta(\Delta t_{i,j}, m_{i,j} \mid \mathcal{H}_{i,<j})$$

The probability is computed by integrating the intensity function $\lambda(t)$ on each time interval. However, this parametrization is sensitive to simultaneous events where the inter-event interval $\Delta t \approx 0$, causing the intensity function to become extremely unstable during pretraining. To address this, MOTOR (Steinberg et al., 2024) separately models the next event of each code and computes the mean likelihood over all codes. Both methods only model non-numeric codes and ignore the value associated with numerical events.

Marked Temporal Point Process (MTPP). Two most recent EHR Foundation models approximate the joint likelihood with the marked temporal point process (Gadd et al., 2025) (Burger et al., 2025). SurvivEHR (Gadd et al., 2025) uses competing risk to model the inter-event time and fit a code-specific Gaussian distribution for numerical codes with the

following likelihood:

$$\mathcal{L}_{\text{SurvivEHR}}(\theta) := \prod_{i=1}^{N} \prod_{j} p_{\theta_1}(\Delta t_{i,j} \mid m_{i,j}, \mathcal{H}_{i,<j}) p_{\theta_2}(v_{i,j} \mid m_{i,j}, \mathcal{H}_{i,<j})$$

Similar to TPP, this method struggles with concurrent events. Additionally, it assumes conditional independence between time and value, thereby biasing the joint likelihood. ICareFM (Burger et al., 2025) addresses these limitations by defining a surrogate loss function for a set of abnormal events K where the numerical codes have exceptionally low or high values, equivalent to maximizing the joint likelihood of all abnormal events:

$$\mathcal{L}_{\text{ICareFM}}(\theta) := \prod_{i=1}^{N} \prod_{(m_{i,j}, v_{i,j}) \in K} p_{\theta}(\Delta t_{i,j} \mid \mathcal{H}_{i,<j})$$

This method is exclusive to numerical measurement and ignores other nonnumerical codes like diagnosis and procedures.

Modeling the complete joint likelihood in Equation (1) is challenging since we need to capture the complex relationship between time, code, and value. EHR events also co-occur frequently, making it hard to fit the intensity function for a standard point process.

D. Comparison of Different Pretraining Losses

Table 1. **Comparison of pretraining objectives of existing EHR foundation models** "Mixed Event Type" denotes if the model includes events with and without associated numerical values in the loss function. "Time–Value Dependence" means that the model considers the dependence between time and the value of the same event. "Concurrent Events" denotes the ability to predict co-occurring events.

Model	Mixed Event Types	Time–Value Dependence	Concurrent Events
Context Clue (Wornow et al., 2024)	✓	✗	✓
MOTOR (Steinberg et al., 2024)	✗	✗	✓
SurvivalEHR (Gadd et al., 2025)	✓	✗	✗
ICareFM (Burger et al., 2025)	✗	✓	✓
ORA (ours)	✓	✓	✓

E. Experimental Setting

E.1. Datasets

Table 2. Dataset Split of MIMIC-IV and CUMC

Split Name	**MIMIC-IV**		**CUMC**	
	# Patients	# Events	# Patients	# Events
Training Set	291,702	579,667,440	3,996,578	1,562,316,866
Tuning Set	36463	71,789,378	705,279	273,247,565
Test Set	36462	71,290,747	2,015,082	781,462,164

E.2. Cohort Definition

Binary Classification. We consider 7 binary classification tasks from Pang et al. (2025):

- Predicting three patient outcomes: In-hospital mortality (Mortality), a length of stay longer than 7 days (LOS), and 30-day readmission (Readmission). The detailed inclusion criteria can be found in section 3.2 of the referenced paper.
- Predicting onset of two chronic conditions: Celiac and Metabolically Associated Steatotic Liver Disease (MASLD).

- Predicting two acute conditions: Acute Myocardial Infarction (AMI) and Ischemic Stroke (Stroke).

For each disease, we define a set of at-risk events as cohort inclusion criteria and a set of case events to determine patients' labels. Compared with using a set of ICD codes, this method adds more task difficulty and is more clinically meaningful.

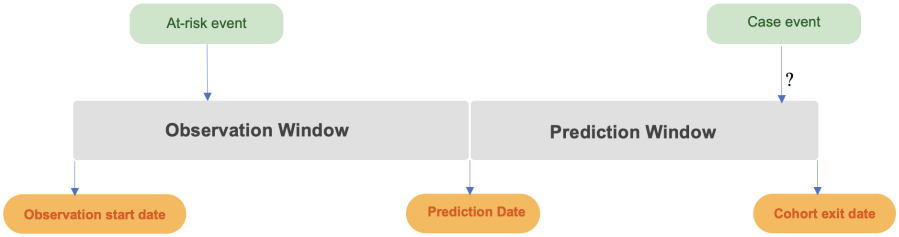

Figure 4. Visualization of cohort definition. The detailed definition of at-risk, prediction time, and case events can be found in Appendix B of Pang et al. (2025).

Predicting chronic and acute conditions is particularly challenging because of the considered 'at-risk' cohort, which increases task difficulty while improving downstream clinical utility (Pang et al., 2025). We treat chronic conditions as incident onset events and therefore evaluate only the first occurrence, consistent with clinical interest in early and first identification. In contrast, acute conditions may recur, so we include all occurrences and formulate a recursive prediction task to capture repeated risk over time.

Time-to-event. Clinical decision-making often depends not only on whether an event will occur, but also on when it is expected to occur. To evaluate this capability, we consider time-to-event prediction for four conditions (Celiac, MASLD, AMI, and Stroke). For each condition, the prediction time is defined at the discharge of an inpatient visit, and the task is to predict the time until the patient receives the first diagnosis of the corresponding condition.

Regression. Comprehensive characterization of patient health requires predicting future physiological measurements. Prior works have largely overlooked regression and TTE as downstream tasks. We assess FM performance on regression tasks involving four important laboratory measurements for sepsis patients: Platelets, Creatinine, Oxygen, and Glucose. Each measurement is identified by a single medical code (see Table 3). All regression tasks are included in the tokenizers' vocabularies for each pretrained model, ensuring they are not out of vocabulary at evaluation time. To avoid leakage across visits, we restrict evaluation to same-visit laboratory tests that occur at least 4 hours after admission, with the prediction time as exactly 4 hours before the corresponding lab event.

Table 3. Corresponding medical codes for all regression tasks on MIMIC and CUMC

Task	MIMIC-IV	CUMC(OMOP)
Creatinine	50912	LOINC/2160-0
Platelets	51265	LOINC/26515-7
Oxygen	50821 (PaO_2)	LOINC/2708-6 (SpO_2)
Glucose	50931	LOINC/2339-0

E.3. Model

E.3.1. EHR Tokenizaion

Following Steinberg et al. (2024), we use a hierarchical tokenizer. For each event, we construct its embedding by summing up the embeddings of each associated attribute: e.g., code, numerical value, etc. This joint event encoding has been demonstrably better than factorizing the event into multiple tokens (Guo et al., 2026). All numerical values are pooled together and discretized into shared quantiles. During tokenization, each continuous value is assigned a quantile token for its value embedding. For codes, we adopt an entropy-based filter to construct the vocabulary with the most informative events (Steinberg et al., 2024). Define $p(m)$ as the probability that the code appears in each patient. The entropy calculation is as follows:

$$H(m) = -p(m)\log p(m)$$

If the dataset has an ontology mapping, we can calculate the conditional entropy of any code m relative to its parent n. Suppose $p(m, n^+)$ denotes the probability that both m and n appear per patient and $p(m, n^-)$ be the probability that only m appears in the patient. $p(m) = p(m, n^+) + p(m, n^-)$, the conditional entropy is as follows:

$$H(m|n) = -p(m, n^-)\log\frac{p(m, n^-)}{p(m)} - p(m, n^+)\log\frac{p(m, n^+)}{p(m)}$$

We quantify the informativeness of each code using its empirical entropy, and retain codes with the highest entropy to reduce noise during learning.

E.3.2. BACKBONE

Following Steinberg et al. (2024), and maintaining consistency to demonstrate the gains from the choice of pre-training loss, we adopt a decoder-only Transformer backbone that avoids look-ahead from future medical events through a causal attention mechanism. Unlike the standard Transformer, it uses rotary position embeddings with age for improved temporality processing. It also adopts local attention and sample packing for efficient pretraining. To demonstrate the generalizability of our pretraining loss, we also evaluate a Mamba-base, a state-space model (Gu & Dao, 2024), as an alternative backbone. Mamba replaces attention-based modules in the Transformer with a selective state-space block. Prior work has shown its advantage over the Transformer in capturing long, irregular EHR sequences with temporal dependencies (Wornow et al., 2024; Fallahpour et al., 2024). For a fair comparison across different models, we set the parameter size for all models to around 120M. Within each architecture, we also use the same configuration files for different losses, which is specified as follows:

Table 4. Model configurations.

Model	**Configuration**	**Value**
Transformer	context length	8192
	learning rate	1e−5
	dim model	768
	intermedicate size	3072
	num layers	11
	num heads	12
	Total Parameters	**119M**
Mamba	context length	8192
	learning rate	2e−4
	dim model	768
	intermedicate size	1536
	num layers	28
	num heads	16
	Total Parameters	**120M**

E.3.3. EFFICIENT PREDICTION HEAD

The last layer of Transformer and Mamba produces an embedding matrix $X \in R^{\times D}$. We need to project it into the final probability matrix $P(x)$ for both nonnumerical codes (eg, diagnosis) and numerical codes with continuous values (eg, lab test). As mentioned in the main paper, we adopt a two-stage projection to be parameter efficient. D indicates the embedding dimension after Transformer or Mamba. D_2 is the hidden dimension after the first projection layer. M_{non} and M_{num} represent nonnumerical codes and numerical codes for pretraining, respectively. The visualization is as follows:

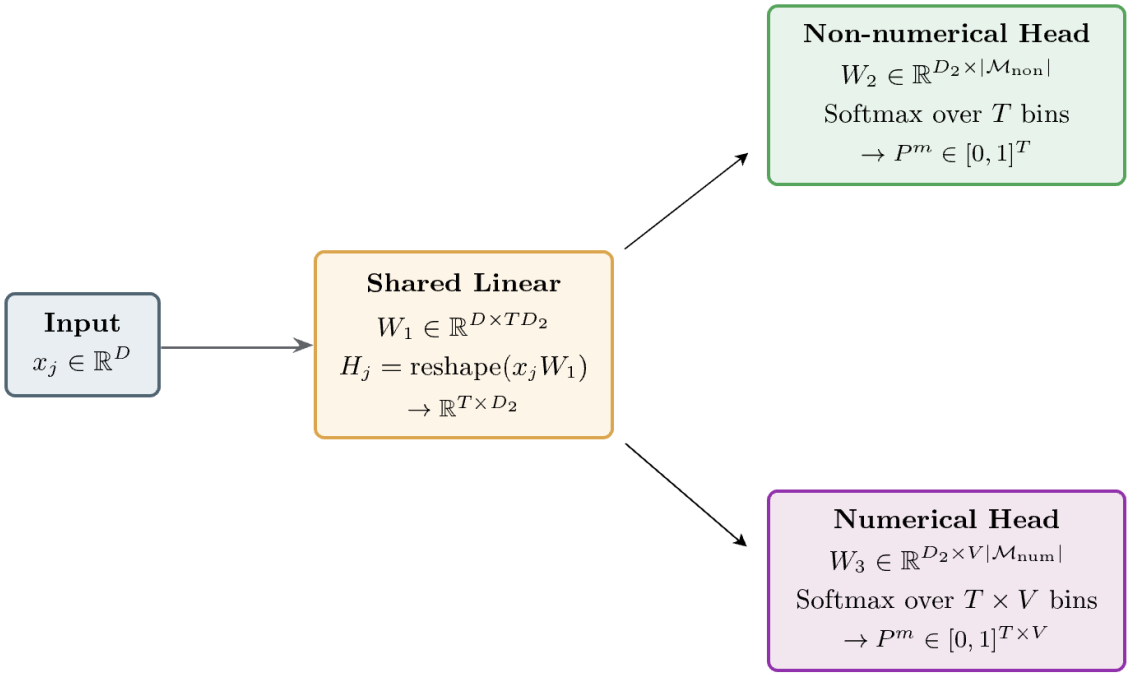

Figure 5. Efficient prediction head: We share the same projection layer for both nonnumerical and numerical codes and then use two different projection heads to output the separate probability matrices.

E.4. Interpretation of Discretized ORA Likelihood

For observed events, this objective maximizes the probability mass assigned to the quantile containing the time and value of the next event, which is equivalent to a standard cross-entropy loss. For censored events, it enforces a low probability of observing any event before the end of the observation window, marginalizing over all possible values. The number of time and value bins, T and V, are shared hyperparameters, while the corresponding quantile boundaries are computed separately for each code from the empirical distributions of inter-event times and observed values. Thus, different codes share the same output dimensionality for the probability matrix but have code-specific discretization intervals, allowing the bins to adapt to heterogeneous event frequencies and value ranges. Additional details and ablation results on the choice of T and V are provided in Appendix E.5.

E.5. Hyperparameter Search for Discretized Intervals

Following prior choices for time and value discretization (Renc et al., 2024; Steinberg et al., 2024), we initially set $T = 8$ and $V = 10$. We further conducted ablations over $T \in \{8, 16\}$ and $V \in \{5, 10\}$ by pretraining four models on each dataset and evaluating them across all downstream tasks. For each task, we ranked the four settings by performance (from best to worst) and reported the average rank across tasks for each setting. We report the ablation study results on MIMIC-IV in Table 5. $T = 8, V = 5$ is the strongest and most stable setting, achieving the lowest average rank. Although coarser discretization uses fewer bins and performs better in downstream evaluation, it may be less precise for capturing fine-grained values during zero-shot generation. In contrast, using more bins increases the number of parameters in the final projection head and may require longer training for stable convergence.

Table 5. Ablation study of time and value discretization on MIMIC-IV. We report the average rank across all downstream tasks. Lower rank reflects better performance.

Setting	Average rank
$T = 8,\ V = 5$	1.33
$T = 16,\ V = 10$	2.58
$T = 16,\ V = 5$	2.75
$T = 8,\ V = 10$	3.33

F. Averaged Performance for Mamba Backbones

To test whether our loss function is robust across different backbones, we further pre-trained all models with the same Mamba backbone and evaluated them on the MIMIC-IV datasets. We present the full comparison of all models in Figure 6. Similarly, we also fix the tokenizer and the backbone to compare how different loss functions impact the downstream task. The results are reported in Figure 7.

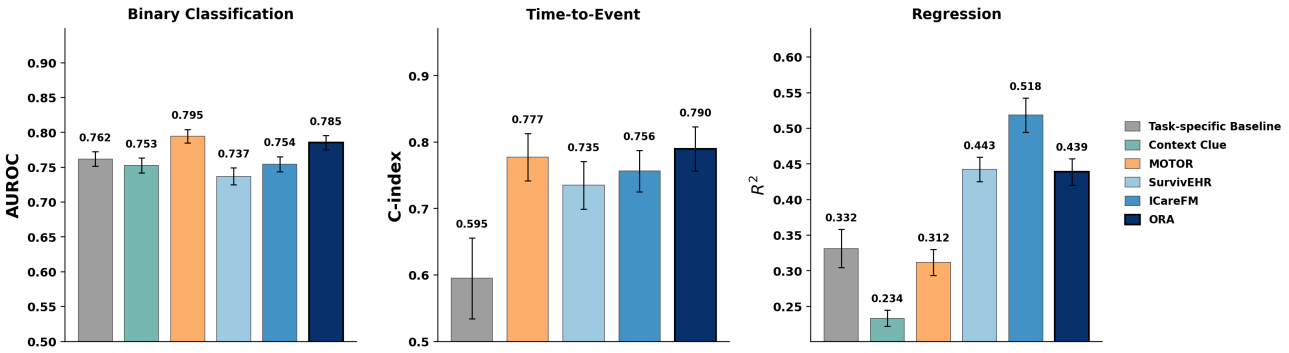

Figure 6. **ORA Improves Generalization Across Classification, Time-to-Event, and Regression Tasks.** Average performance on MIMIC-IV shows that jointly modeling event timing and values yields consistently stronger representations. ORA outperforms different foundation models across both binary classification and TTE tasks and remains competitive in regression results. All models used the **Mamba** backbone.

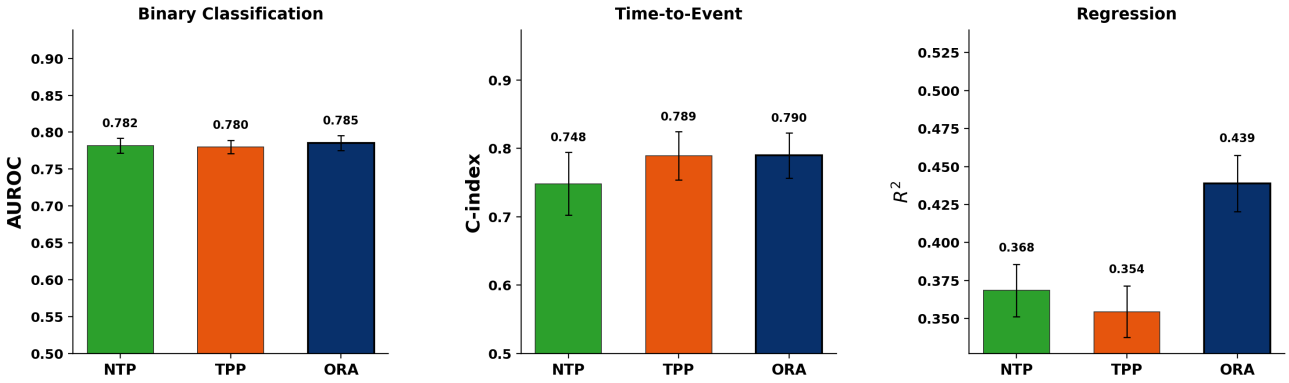

Figure 7. **Pretraining Objective Drives Downstream Performance Across Task Families on MIMIC-IV using the Mamba Backbone.** Under a fixed tokenizer and Mamba backbone, jointly modeling time-to-event and values yields consistent gains over NTP and TPP objectives, demonstrating that improvements arise from the loss formulation rather than model capacity or input representation.

G. Full Experiment results

This section details the performance for each backbone, loss, and task across both datasets.

G.1. MIMIC-IV

We evaluate all models on 6 classification tasks, 3 time-to-event tasks, and 4 regression tasks on MIMIC-IV. Note that we don't extract the Celiac cohort in MIMIC-IV.

G.1.1. Classification

Table 6. AUROC of binary classification tasks for MIMIC-IV patients. *Larger is better.*

Model	Readmission	LOS	Mortality	AMI	MASLD	Stroke
XGBoost	0.726 (0.003)	0.748 (0.003)	0.882 (0.005)	0.807 (0.008)	0.700 (0.020)	0.708 (0.023)
CC-Llama	0.730 (0.003)	0.781 (0.003)	0.901 (0.005)	0.796 (0.010)	0.663 (0.020)	0.696 (0.025)
CC-Mamba	0.731 (0.003)	0.779 (0.003)	0.896 (0.006)	0.783 (0.009)	0.661 (0.019)	0.665 (0.024)
MOTOR	0.743 (0.002)	0.833 (0.002)	0.954 (0.002)	0.825 (0.008)	0.703 (0.021)	0.710 (0.019)
Transformer-NTP	0.728 (0.003)	0.808 (0.002)	0.926 (0.004)	0.810 (0.008)	0.691 (0.020)	0.691 (0.025)
Transformer-TPP	0.744 (0.004)	0.839 (0.002)	0.963 (0.003)	0.818 (0.007)	0.705 (0.018)	0.700 (0.020)
Transformer-SurvivEHR	0.714 (0.003)	0.734 (0.003)	0.898 (0.005)	0.803 (0.007)	0.682 (0.016)	0.603 (0.022)
Transformer-ICareFM	0.719 (0.004)	0.796 (0.003)	0.935 (0.005)	0.802 (0.007)	0.659 (0.021)	0.652 (0.023)
Transformer-ORA	0.745 (0.003)	**0.841 (0.002)**	**0.965 (0.002)**	0.827 (0.007)	0.714 (0.017)	0.711 (0.018)
Mamba-NTP	0.732 (0.004)	0.813 (0.002)	0.933 (0.004)	0.814 (0.008)	0.679 (0.020)	**0.719 (0.022)**
Mamba-TPP	0.741 (0.003)	0.809 (0.003)	0.934 (0.004)	0.822 (0.007)	0.709 (0.020)	0.666 (0.017)
Mamba-SurvivEHR	0.714 (0.004)	0.743 (0.003)	0.900 (0.006)	0.786 (0.007)	0.643 (0.021)	0.635 (0.029)
Mamba-ICareFM	0.728 (0.003)	0.787 (0.003)	0.931 (0.004)	0.792 (0.007)	0.661 (0.021)	0.627 (0.024)
Mamba-ORA	**0.747 (0.003)**	0.812 (0.002)	0.939 (0.004)	**0.832 (0.007)**	**0.722 (0.020)**	0.660 (0.022)

G.1.2. TIME-TO-EVENT

Table 7. C-index of time-to-event prediction tasks for MIMIC-IV patients. *Larger is better.*

Model	AMI	MASLD	Stroke
Deephit	0.651 (0.029)	0.560 (0.024)	0.574 (0.085)
CC-Llama	0.810 (0.029)	0.688 (0.022)	0.843 (0.038)
MOTOR	0.811 (0.030)	0.689 (0.020)	0.832 (0.041)
Transformer-NTP	0.796 (0.030)	0.706 (0.018)	0.702 (0.056)
Transformer-TPP	**0.856 (0.028)**	0.710 (0.020)	0.886 (0.030)
Transformer-SurvivEHR	0.756 (0.032)	0.667 (0.019)	0.832 (0.044)
Transformer-ICareFM	0.757 (0.035)	0.641 (0.025)	0.809 (0.042)
Transformer-ORA	0.833 (0.030)	**0.723 (0.020)**	**0.890 (0.027)**
Mamba-NTP	0.808 (0.029)	0.673 (0.018)	0.764 (0.062)
Mamba-TPP	0.851 (0.028)	0.698 (0.020)	0.819 (0.042)
Mamba-SurvivEHR	0.779 (0.028)	0.648 (0.018)	0.778 (0.044)
Mamba-ICareFM	0.776 (0.028)	0.668 (0.019)	0.825 (0.034)
Mamba-ORA	**0.856 (0.032)**	0.664 (0.020)	0.849 (0.034)

G.1.3. REGRESSION

Table 8. R^2 of lab test regression for MIMIC-IV patients. *Larger coefficient reflects a larger proportion of variance explained.*

Model	Creatinine	PaO2	Platelets	Glucose
CC-Llama	0.319 (0.018)	0.219 (0.005)	0.290 (0.004)	0.155 (0.015)
CC-Mamba	0.307 (0.017)	0.218 (0.005)	0.261 (0.004)	0.148 (0.014)
MOTOR	0.556 (0.033)	0.231 (0.005)	0.298 (0.004)	0.163 (0.016)
Most Recent	**0.877 (0.034)**	-0.431 (0.024)	**0.910 (0.002)**	-0.030 (0.035)
Transformer-NTP	0.489 (0.025)	0.259 (0.005)	0.312 (0.004)	0.180 (0.017)
Transformer-TPP	0.603 (0.030)	0.234 (0.005)	0.310 (0.003)	0.154 (0.015)
Transformer-SurvivEHR	0.607 (0.033)	0.185 (0.004)	0.437 (0.004)	0.196 (0.019)
Transformer-ICareFM	0.784 (0.044)	0.284 (0.006)	0.844 (0.002)	0.134 (0.019)
Transformer-ORA	0.603 (0.028)	0.267 (0.005)	0.605 (0.003)	0.186 (0.018)
Mamba-NTP	0.595 (0.029)	0.270 (0.005)	0.407 (0.003)	**0.202 (0.019)**
Mamba-TPP	0.617 (0.030)	0.241 (0.005)	0.400 (0.004)	0.160 (0.016)
Mamba-SurvivEHR	0.693 (0.029)	0.182 (0.004)	0.705 (0.003)	0.190 (0.019)
Mamba-ICareFM	0.783 (0.046)	0.282 (0.006)	0.855 (0.003)	0.154 (0.015)
Mamba-ORA	0.645 (0.034)	**0.286 (0.006)**	0.659 (0.003)	0.166 (0.015)

Table 9. RMSE of lab test regression for MIMIC-IV patients. *Lower is better.*

Model	Creatinine	PaO2	Platelets	Glucose
Baseline	**0.503 (0.084)**	77.727 (0.535)	**38.632 (0.326)**	69.773 (3.440)
CC-Llama	1.244 (0.053)	57.369 (0.466)	108.897 (0.653)	62.800 (3.797)
CC-Mamba	1.255 (0.052)	57.421 (0.462)	111.113 (0.635)	63.067 (3.774)
MOTOR	0.968 (0.067)	56.920 (0.447)	108.285 (0.630)	62.684 (3.672)
Transformer-NTP	1.038 (0.058)	55.888 (0.461)	107.198 (0.647)	62.017 (3.684)
Transformer-TPP	0.915 (0.063)	56.816 (0.452)	107.359 (0.605)	63.011 (3.641)
Transformer-SurvivEHR	0.911 (0.067)	58.621 (0.467)	96.957 (0.584)	61.416 (3.713)
Transformer-ICareFM	0.673 (0.090)	54.937 (0.430)	51.088 (0.529)	63.730 (3.543)
Transformer-ORA	0.915 (0.062)	55.576 (0.439)	81.202 (0.593)	61.753 (3.700)
Mamba-NTP	0.925 (0.062)	55.475 (0.451)	99.533 (0.632)	**61.199 (3.721)**
Mamba-TPP	0.899 (0.064)	56.549 (0.447)	100.106 (0.567)	62.764 (3.671)
Mamba-ORA	0.866 (0.068)	**54.877 (0.429)**	75.476 (0.581)	62.572 (3.647)

Table 10. MAE of lab test regression for MIMIC-IV patients. *Lower is better.*

Model	Creatinine	PaO2	Platelets	Glucose
Baseline	**0.193 (0.002)**	45.537 (0.324)	**25.184 (0.136)**	33.951 (0.272)
CC-Llama	0.803 (0.004)	38.620 (0.230)	78.139 (0.312)	34.944 (0.251)
CC-Mamba	0.811 (0.004)	38.593 (0.219)	80.145 (0.317)	35.188 (0.255)
MOTOR	0.542 (0.004)	38.211 (0.206)	77.215 (0.322)	34.396 (0.205)
Transformer-NTP	0.651 (0.004)	37.102 (0.218)	76.247 (0.310)	34.080 (0.200)
Transformer-TPP	0.507 (0.003)	38.114 (0.217)	76.565 (0.310)	34.745 (0.212)
Transformer-SurvivEHR	0.522 (0.003)	38.938 (0.214)	69.180 (0.279)	**33.071 (0.190)**
Transformer-ICareFM	0.350 (0.003)	36.794 (0.198)	34.963 (0.182)	36.130 (0.209)
Transformer-ORA	0.504 (0.003)	36.859 (0.209)	56.163 (0.249)	33.630 (0.204)
Mamba-NTP	0.547 (0.003)	36.983 (0.209)	70.657 (0.284)	33.184 (0.194)
Mamba-TPP	0.492 (0.003)	37.667 (0.207)	70.825 (0.277)	34.357 (0.202)
Mamba-ORA	0.448 (0.003)	**36.319 (0.199)**	51.451 (0.236)	34.393 (0.202)

G.2. CUMC

As external validation, this section shows similar results across tasks and backbones on the CUMC dataset, a large urban center.

G.2.1. CLASSIFICATION

Table 11. AUROC of binary classification tasks for CUMC patients. *Larger is better.*

Model	Readmission	LOS	Mortality	AMI	MASLD	Stroke	Celiac
XGBoost	0.732 (0.003)	0.773 (0.002)	0.875 (0.005)	0.833 (0.008)	0.681 (0.009)	0.872 (0.006)	0.650 (0.031)
Llama	0.758 (0.003)	0.798 (0.002)	0.892 (0.004)	0.821 (0.009)	0.731 (0.008)	0.846 (0.007)	0.604 (0.040)
Mamba	0.757 (0.003)	0.784 (0.002)	0.883 (0.004)	0.816 (0.009)	0.706 (0.008)	0.865 (0.007)	0.635 (0.038)
MOTOR	0.783 (0.003)	0.850 (0.002)	0.963 (0.002)	0.853 (0.007)	0.727 (0.007)	0.869 (0.006)	0.647 (0.030)
Transformer-NTP	0.760 (0.003)	0.842 (0.002)	0.938 (0.003)	0.821 (0.008)	0.713 (0.009)	0.849 (0.007)	0.624 (0.039)
Transformer-TPP	0.784 (0.003)	0.857 (0.002)	0.963 (0.002)	0.843 (0.008)	0.718 (0.008)	0.863 (0.006)	0.736 (0.033)
Transformer-SurvivEHR	0.749 (0.003)	0.795 (0.002)	0.923 (0.004)	0.826 (0.008)	0.709 (0.008)	0.839 (0.006)	0.629 (0.031)
Transformer-ICareFM	0.760 (0.003)	0.846 (0.002)	0.948 (0.003)	0.825 (0.008)	0.702 (0.008)	0.828 (0.007)	0.594 (0.035)
Transformer-ORA	**0.787 (0.003)**	**0.866 (0.002)**	**0.968 (0.002)**	0.850 (0.008)	0.745 (0.007)	0.870 (0.007)	**0.749 (0.036)**
Mamba-NTP	0.757 (0.003)	0.817 (0.002)	0.915 (0.003)	0.780 (0.010)	0.681 (0.008)	0.830 (0.007)	0.637 (0.033)
Mamba-TPP	0.776 (0.003)	0.859 (0.002)	0.942 (0.003)	0.854 (0.008)	0.737 (0.008)	0.864 (0.006)	0.739 (0.034)
Mamba-ORA	0.779 (0.002)	0.861 (0.002)	0.946 (0.003)	**0.858 (0.008)**	**0.766 (0.008)**	**0.873 (0.006)**	0.744 (0.038)

G.2.2. TIME-TO-EVENT

Table 12. C-index of time-to-event prediction tasks for CUMC patients. *Larger is better.*

Model	AMI	Celiac	Stroke	MASLD
Deephit	0.615 (0.008)	0.495 (0.029)	0.637 (0.008)	0.602 (0.006)
CC-Llama	0.758 (0.011)	0.622 (0.034)	0.808 (0.009)	0.704 (0.009)
CC-Mamba	0.774 (0.011)	0.693 (0.036)	0.815 (0.009)	0.692 (0.009)
MOTOR	0.792 (0.012)	0.645 (0.036)	0.804 (0.009)	0.674 (0.009)
Transformer-NTP	0.724 (0.012)	0.564 (0.035)	0.769 (0.010)	0.637 (0.009)
Transformer-TPP	0.775 (0.014)	0.694 (0.031)	0.795 (0.010)	0.664 (0.010)
Transformer-SurvivEHR	**0.821 (0.009)**	0.681 (0.033)	0.810 (0.010)	0.700 (0.009)
Transformer-ICareFM	0.767 (0.009)	0.667 (0.028)	0.742 (0.011)	0.667 (0.010)
Transformer-ORA	0.792 (0.013)	**0.743 (0.029)**	0.793 (0.010)	0.692 (0.011)
Mamba-NTP	0.730 (0.011)	0.507 (0.043)	0.782 (0.009)	0.669 (0.008)
Mamba-TPP	0.767 (0.012)	0.651 (0.034)	**0.823 (0.010)**	**0.708 (0.010)**
Mamba-ORA	0.785 (0.012)	0.725 (0.038)	0.788 (0.011)	0.704 (0.009)

G.2.3. REGRESSION

Table 13. R^2 of lab test regression for CUMC patients. *Larger coefficient reflects a larger proportion of variance explained.*

Model	Creatinine	Platelets	SpO2	Glucose
Most Recent	0.679 (0.036)	0.426 (0.011)	0.078 (0.013)	0.052 (0.015)
CC-Llama	0.298 (0.013)	0.150 (0.005)	0.770 (0.007)	0.217 (0.007)
CC-Mamba	0.277 (0.013)	0.146 (0.006)	0.762 (0.007)	0.196 (0.006)
MOTOR	0.469 (0.020)	0.198 (0.007)	0.842 (0.006)	0.282 (0.008)
Transformer-NTP	0.431 (0.022)	0.228 (0.006)	0.816 (0.007)	0.287 (0.009)
Transformer-TPP	0.397 (0.018)	0.203 (0.007)	0.832 (0.006)	0.279 (0.008)
Transformer-SurvivEHR	0.534 (0.024)	0.481 (0.007)	0.798 (0.006)	0.365 (0.009)
Transformer-ICareFM	**0.708 (0.031)**	**0.568 (0.008)**	0.846 (0.006)	**0.431 (0.009)**
Transformer-ORA	0.542 (0.024)	0.349 (0.008)	0.836 (0.006)	0.305 (0.008)
Mamba-NTP	0.423 (0.024)	0.220 (0.007)	0.809 (0.006)	0.208 (0.007)
Mamba-TPP	0.543 (0.027)	0.311 (0.008)	0.849 (0.006)	0.306 (0.008)
Mamba-ORA	0.582 (0.029)	0.464 (0.007)	**0.852 (0.006)**	0.330 (0.008)

Table 14. RMSE of lab test regression for CUMC patients. *Lower is better.*

Model	Creatinine	Platelets	SpO2	Glucose
Most Recent	0.630 (0.047)	74.419 (1.048)	15.598 (0.225)	68.571 (0.78)
CC-Llama	0.987 (0.035)	90.486 (0.941)	7.802 (0.114)	62.172 (0.703)
CC-Mamba	1.001 (0.035)	90.695 (0.939)	7.939 (0.112)	63.002 (0.742)
MOTOR	0.825 (0.042)	88.308 (0.948)	6.437 (0.123)	59.754 (0.624)
Transformer-NTP	0.856 (0.039)	86.837 (0.969)	6.948 (0.110)	59.545 (0.580)
Transformer-TPP	0.879 (0.041)	88.046 (0.936)	6.638 (0.117)	59.888 (0.616)
Transformer-SurvivEHR	0.772 (0.043)	71.032 (0.861)	7.289 (0.117)	56.187 (0.590)
Transformer-ICareFM	**0.612 (0.049)**	**64.829 (0.838)**	6.354 (0.122)	**53.214 (0.546)**
Transformer-ORA	0.766 (0.044)	79.574 (0.883)	6.555 (0.122)	58.804 (0.609)
Mamba-NTP	0.860 (0.045)	87.097 (0.881)	7.078 (0.110)	62.749 (0.663)
Mamba-TPP	0.765 (0.047)	81.869 (0.911)	6.301 (0.123)	58.752 (0.609)
Mamba-ORA	0.732 (0.047)	72.188 (0.878)	**6.226 (0.118)**	57.703 (0.592)

Table 15. MAE of lab test regression for CUMC patients. *Lower is better.*

Model	Creatinine	Platelets	SpO2	Glucose
Most Recent	**0.235 (0.004)**	47.591 (0.428)	6.377 (0.112)	45.428 (0.36)
CC-Llama	0.567 (0.006)	65.073 (0.458)	4.670 (0.046)	42.073 (0.322)
CC-Mamba	0.567 (0.006)	65.327 (0.488)	4.742 (0.045)	42.848 (0.350)
MOTOR	0.402 (0.006)	63.290 (0.478)	3.325 (0.038)	39.864 (0.289)
Transformer-NTP	0.445 (0.005)	62.540 (0.436)	3.968 (0.037)	40.257 (0.299)
Transformer-TPP	0.406 (0.006)	63.068 (0.435)	3.448 (0.037)	39.893 (0.274)
Transformer-SurvivEHR	0.367 (0.005)	48.594 (0.339)	4.145 (0.045)	36.864 (0.267)
Transformer-ICareFM	0.275 (0.004)	**43.674 (0.323)**	3.253 (0.040)	**35.178 (0.249)**
Transformer-ORA	0.341 (0.005)	56.882 (0.376)	3.377 (0.040)	38.962 (0.287)
Mamba-NTP	0.434 (0.006)	62.555 (0.393)	4.086 (0.039)	42.080 (0.286)
Mamba-TPP	0.340 (0.005)	58.465 (0.407)	3.196 (0.038)	38.868 (0.282)
Mamba-ORA	0.326 (0.005)	49.793 (0.357)	**3.103 (0.037)**	37.976 (0.276)