TABEHR: BRIDGING LONGITUDINAL EHRS AND TABULAR FOUNDATION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Longitudinal electronic health records (EHRs) are heterogeneous and irregular: patients differ in which clinical concepts are recorded, when they are observed, and how often they occur, and these recording patterns also vary across institutions. This makes it difficult to reuse clinical prediction models across sites. Tabular foundation models (TFMs) offer reusable in-context prediction from labeled support patients without parameter updates, but require corresponding feature columns that irregular EHR histories do not provide. We introduce **TabEHR**, which learns this missing interface by mapping irregular histories into shared semantic columns for a frozen TFM. TabEHR constructs clinically enriched concept embeddings from ontology-informed descriptions synthesized through four complementary clinical perspectives and uses their geometry to define fixed semantic anchors that aggregate patient events into column-specific embeddings. Keeping the anchors and TFM fixed, we train TabEHR on MIMIC-IV to learn how patient events populate these shared columns. TabEHR achieves the highest macro-averaged AUROC and AUPRC across nine tasks among the evaluated methods, both on held-out MIMIC-IV patients and on the external EHRSHOT cohort, without using EHRSHOT outcome labels.

1 INTRODUCTION

Clinical prediction from longitudinal electronic health records (EHRs) uses observations available before a prediction time to estimate a patient's clinical outcomes (Choi et al., 2016b; Rajkomar et al., 2018). These records interleave diagnoses, medications, procedures, and numerical measurements, and patient histories vary substantially in length, observation density, and temporal spacing (Rajkomar et al., 2018; Lasko et al., 2013). Across institutions, differences in recorded concept vocabularies add another source of variation, making patient representations learned from one EHR system difficult to reuse in another (Kim et al., 2025b).

To model these irregular longitudinal histories, EHR sequence models learn patient representations from clinical events over time. Recurrent attention and pretrained Transformers model dependencies across visits and events (Choi et al., 2016a; Li et al., 2020; Rasmy et al., 2021), while clinical language modeling learns reusable patient representations by predicting subsequent observations (Steinberg et al., 2021). This line of research has expanded to include multiple clinical domains (Kim et al., 2025a), quantitative measurements (Du et al., 2026), and architectures that can use longer patient histories (Li et al., 2023; Fallahpour et al., 2025; Wornow et al., 2025). These approaches model a patient's observed history, but representing the clinical meaning of individual concepts remains important, particularly when coding vocabularies differ across institutions.

A separate line of work enriches the meaning of individual clinical concepts using knowledge beyond their occurrence in patient records. Ontologies provide relational structure among concepts (Choi et al., 2017), while code descriptions encoded with pretrained biomedical language models yield textual representations (Hur et al., 2022; Gu et al., 2020). Hybrid approaches combine textual and relational information to learn concept embeddings or discrete medical tokens (Kim et al., 2025b; Su et al., 2025). Medical knowledge can also guide patient-level aggregation through personalized graphs built from the knowledge associated with each patient's record concepts (Jiang et al., 2024). Together with advances in sequence modeling, these methods improve the representation of both longitudinal histories and the concepts they contain.

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.

Representation reuse, however, does not by itself make the predictor reusable across clinical outcomes. Conventional clinical prediction pipelines typically train a task-specific predictor for each outcome of interest. In-context tabular foundation models (TFMs) offer a different form of reuse: they condition predictions on labeled support examples supplied at inference, allowing the same predictor to serve different tasks without updating its parameters (Jäger et al., 2026; Qu et al., 2026; Eo et al., 2026). Applying this mechanism to longitudinal EHRs requires corresponding feature columns across support and query examples, whereas patient records contain variable sets of clinical events. Existing clinical applications construct these columns through predefined aggregation, including summaries of past measurements and observations within fixed time windows (Ding et al., 2025; Pham et al., 2026b). Building on these features, retrieval-based approaches learn to select relevant support patients and, in adapted variants, map the inputs to a frozen predictor (Pham et al., 2026b). These approaches learn how patient features are used for prediction, while the aggregation of individual events into those features remains specified in advance.

We introduce **TabEHR**, which learns this missing interface by transforming irregular clinical histories into shared semantic columns for a frozen TFM. To provide a clinically meaningful basis for this interface, we construct concept embeddings from descriptions that integrate each OMOP concept's name and typed ontology relations through four complementary clinical perspectives elicited through distinct expert-role prompts, and refine the embedding space using masked-language modeling and concept-level contrastive learning. TabEHR uses the resulting semantic geometry to construct a fixed set of *semantic anchors* that define common column identities across patients. Patient events are encoded using their concept embeddings, numerical values, and recency, then aggregated by the anchors into patient-specific readouts that serve as cell embeddings for the TFM. The clinical prediction loss trains the event encoder and anchor aggregator jointly across source tasks through the frozen TFM, while the concept embeddings and semantic anchors remain fixed.

We train TabEHR on MIMIC-IV (Johnson et al., 2023) and evaluate the same learned parameters across nine clinical prediction tasks on held-out MIMIC-IV patients and the external EHRSHOT cohort (Wornow et al., 2023). Both evaluations use MIMIC-IV training patients as support, without using EHRSHOT outcome labels for prediction. TabEHR achieves the highest macro-averaged AUROC and AUPRC among the evaluated methods on both cohorts (Table 1). On EHRSHOT, it outperforms predefined statistical features supplied to the same TFM, supporting the value of the learned patient representation for cross-institution prediction.

Our contributions are:

- We introduce **TabEHR**, which learns to transform irregular longitudinal EHRs into shared feature columns for tabular in-context prediction. Patient-specific readouts enter a frozen TFM directly as cell embeddings, allowing event encoding and aggregation to be learned through the clinical prediction objective.
- We develop concept embeddings from ontology-informed clinical descriptions and use them both to represent events and to construct fixed semantic anchors that define corresponding columns across patients. Concept retrieval analysis shows that the resulting anchor space preserves established clinical relationships across four clinical domains.
- TabEHR achieves the highest macro-averaged AUROC and AUPRC across nine tasks among the evaluated methods on both held-out MIMIC-IV patients and the external EHRSHOT cohort, without using EHRSHOT outcome labels for prediction. Its external gains over predefined statistical features supplied to the same TFM support the value of learning patient representations for cross-institution transfer.

2 Related Work

2.1 EHR sequence models

Sequence models for EHRs represent a patient's history as a sequence of clinical events or visits and learn dependencies across observations. Recurrent attention models weight visits and clinical variables (Choi et al., 2016a), while pretrained Transformers learn contextual representations from code sequences (Li et al., 2020; Rasmy et al., 2021). Clinical language modeling also learns reusable patient representations by predicting subsequent observations (Steinberg et al., 2021). Other work

broadens what these sequences contain, incorporating multiple clinical domains (Kim et al., 2025a) and quantitative laboratory values (Du et al., 2026). Longer histories are addressed through hierarchical attention (Li et al., 2023) and state-space architectures (Fallahpour et al., 2025), alongside systematic studies of context length and model architecture (Wornow et al., 2025).

2.2 Concept representations and medical knowledge

Medical knowledge informs concept representations for EHRs through ontologies, textual descriptions, and biomedical language pretraining. Ontology-based approaches such as GRAM represent a concept through an attention-weighted combination of its own embedding and those of its ancestors (Choi et al., 2017). Text-based approaches encode code descriptions in a shared language space (Hur et al., 2022), while biomedical language pretraining provides domain-specific text encoders (Gu et al., 2020). Hybrid methods combine textual and relational information, refining embeddings of generated concept descriptions through ontology-based graph learning (Kim et al., 2025b), or quantizing separately encoded text and graph representations into medical tokens (Su et al., 2025). Knowledge can also guide patient-level aggregation: GraphCare constructs a personalized graph by combining the knowledge graphs associated with a patient's recorded concepts, and uses it for prediction (Jiang et al., 2024).

2.3 Tabular representations and in-context prediction

Statistical tabularization converts EHRs into concept counts and windowed summaries, which support strong boosted-tree baselines for clinical prediction (Oufattole et al., 2024; Chen & Guestrin, 2016; Wornow et al., 2023). In-context tabular foundation models (TFMs) offer reusable predictors: pretrained on synthetic tasks sampled from data-generating priors, they condition predictions on labeled training examples supplied as context without updating model parameters at inference (Jäger et al., 2026; Qu et al., 2026; Eo et al., 2026). Applying these predictors to clinical records requires patient features and a prediction context. Longitudinal information can be incorporated through predefined summaries of recent values, recency, and rates of change (Ding et al., 2025). Starting from tabulated clinical snapshots, other work learns task-aligned representations for retrieving support patients and, in adapted variants, an input mapping through a frozen backbone (Pham et al., 2026b). For censored outcomes, survival-aware adaptation trains a prediction head on context-conditioned patient representations (Pham et al., 2026a).

3 Methods

3.1 Problem Setup and Overview

For a clinical task $q \in \mathcal{T}$, a prediction instance i pairs a patient with a prediction time T_i. Its inputs are static covariates $\boldsymbol{x}_i \in \mathbb{R}^{d_x}$ and longitudinal records $\mathcal{H}_i$ observed before T_i, and its target is $y_i \in \mathcal{Y}_q$. A patient may contribute multiple instances at different prediction times.

TabEHR operates on episodes $(q, \mathcal{S}, \mathcal{Q})$, where $\mathcal{S}$ contains labeled support instances and $\mathcal{Q}$ contains query instances, with no patients shared between them. Only support labels are provided to the predictor; query labels are used only for training loss or evaluation.

A tabular foundation model (TFM) requires corresponding feature columns across support and query instances. TabEHR learns this interface from irregular EHRs by using ontology-informed concept embeddings (section 3.2) to construct fixed semantic anchors (section 3.4), each defining a shared column across patients. Figure 1 separates the construction of these shared semantic references (a) from patient-level encoding, aggregation, and prediction (b).

For each instance, concept-wise temporal binning and coverage-first selection produce retained event tokens $\mathcal{E}_i$ (section 3.3). The event encoder and anchor aggregator map these tokens to M patient-specific anchor readouts:

$$\hat{\boldsymbol{Z}}_i = F_\theta(\mathcal{E}_i) = [\hat{\boldsymbol{z}}_{i1}, \ldots, \hat{\boldsymbol{z}}_{iM}]^\top \in \mathbb{R}^{M \times d}. \tag{1}$$

Each anchor defines the same feature column across patients, while its readout $\hat{\boldsymbol{z}}_{im}$ depends on the events observed for instance i.

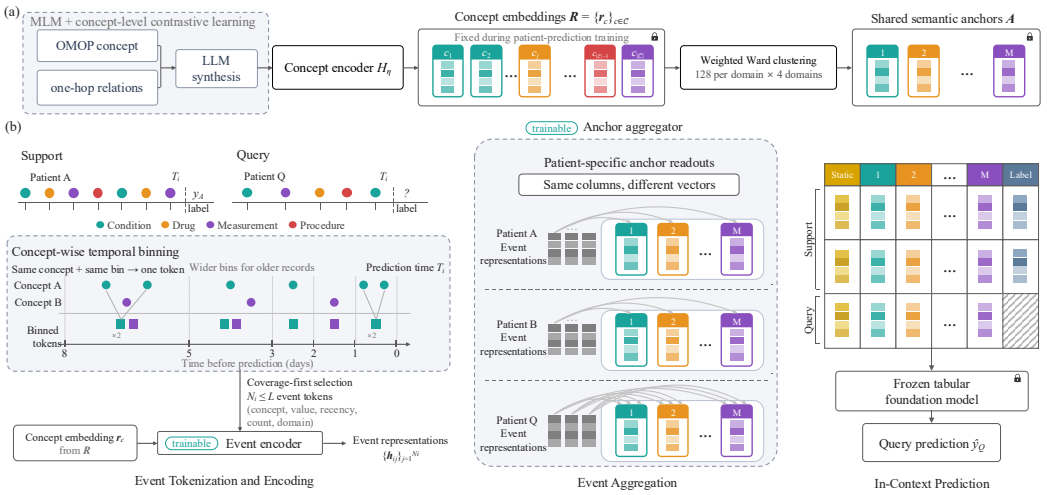

Figure 1: TabEHR overview. (a) Concept embeddings and semantic anchors. (b) Event encoding, anchor aggregation, and in-context prediction. Locks denote components frozen during prediction training.

The anchor readouts are combined with encoded static covariates to form $X_i^{\text{cell}} \in \mathbb{R}^{(M+d_x)\times d}$ and supplied to a frozen TFM g_ϕ (Eo et al., 2026). Given the support labels, the model predicts

$$\hat{y}_i = g_\phi\big(X^{\text{cell}}_{\mathcal{S}\cup\mathcal{Q}}, y_{\mathcal{S}}\big)_i\,, \qquad i \in \mathcal{Q}. \tag{2}$$

During source-task training, the event encoder and anchor aggregator are optimized through the frozen TFM, while the concept embeddings, semantic anchors, and TFM remain fixed. The following sections describe each component and the training procedure in detail.

3.2 Concept Embedding Space

To represent the clinical meaning of recorded concepts, we construct a semantic embedding space from ontology-informed clinical descriptions. For each OMOP concept, the description generator receives its name, domain, vocabulary, and typed one-hop ontology neighborhood. A language model produces four complementary clinical perspectives—an internist, pharmacist, clinical scientist, and informaticist—and integrates them into a single clinical synthesis. Each ontology edge is assessed as valid, outdated, incorrect, or insufficient, and relations judged outdated or incorrect are excluded from the synthesis. The resulting descriptions enrich concept names with local relational evidence. Generation details are provided in Appendix B.1.

We use masked-language modeling (MLM) to learn from the generated clinical descriptions and subsequently combine MLM with a concept-level contrastive objective. For each concept c, let x_c^{syn} denote its integrated synthesis and let $x_c^{(k)}$ denote one of its four clinical perspectives, sampled uniformly during contrastive training. Both texts are encoded and mean-pooled over the non-padding tokens of the final hidden layer. Their ℓ_2-normalized representations are

$$\mathbf{s}_c = \text{norm}(\text{pool}(H_\eta(x_c^{\text{syn}})))\,, \qquad \mathbf{v}_c = \text{norm}\Big(\text{pool}\Big(H_\eta(x_c^{(k)})\Big)\Big)\,, \tag{3}$$

where H_η denotes the concept encoder.

Within a minibatch of B concepts, the synthesis and sampled perspective of the same concept form a positive pair, while representations of other concepts provide in-batch negatives. Let $\mathbf{S}$ and $\mathbf{V}$ stack the normalized synthesis and perspective representations, respectively, and let $\mathbf{y} = (1, \ldots, B)$ denote the matching pair indices. We use the symmetric InfoNCE objective

$$\mathcal{L}_{\text{con}} = \frac{1}{2}\left[\text{CE}\left(\frac{\mathbf{S}\mathbf{V}^\top}{\tau}, \mathbf{y}\right) + \text{CE}\left(\frac{\mathbf{V}\mathbf{S}^\top}{\tau}, \mathbf{y}\right)\right], \tag{4}$$

where τ is the contrastive temperature. During joint training, MLM is applied together with the contrastive objective,

$$\mathcal{L}_{\text{concept}} = \mathcal{L}_{\text{MLM}} + \lambda_{\text{con}}\mathcal{L}_{\text{con}}, \tag{5}$$

where λ_{con} controls the contribution of the contrastive term. The contrastive objective encourages the integrated synthesis and role-specific descriptions of the same clinical concept to occupy nearby regions of the embedding space while distinguishing different concepts.

After concept-level training, each vocabulary entry is encoded once into a fixed vector,

$$\boldsymbol{R} = \{\boldsymbol{r}_c\}_{c\in\mathcal{C}}, \qquad \boldsymbol{r}_c \in \mathbb{R}^{d_r}, \qquad d_r = 768. \tag{6}$$

For concepts with an integrated synthesis, $\boldsymbol{r}_c$ is obtained by mean-pooling the final hidden states over non-padding tokens. Measurement-specific variants are encoded from their corresponding textual descriptions as detailed in Appendix B.1. The resulting concept table remains fixed throughout patient-prediction training and provides the semantic basis for both event representations and the semantic anchors described in section 3.4. The scope of source and target concept metadata is defined in section 4.1. To directly assess the relational structure of this frozen embedding space, we evaluate it without fitting a downstream predictor (Appendix H.1). Across seven clinical relation types, our concept space achieves a mean AUROC of 0.966 for distinguishing related from unrelated concept pairs, compared with 0.848 for MedRep and 0.736 for BiomedBERT. In analogy retrieval, it achieves a mean MRR of 0.275, compared with 0.124 and 0.127, respectively, indicating more consistent relation-specific structure in the learned geometry.

3.3 Event Tokenization and Encoding

We use the concept embeddings to encode events in each patient's history, together with their numerical values, recency, counts, and domains. We first group the available records into event tokens and select a bounded set for encoding.

To construct these tokens, we group records of the same concept by elapsed time before T_i. Records are first consolidated at daily resolution and then merged into bins with geometrically increasing widths. For $\delta = 0.5$, the ordered, deduplicated edge set is

$$\{b_k\}_{k\geq 0} = \operatorname{sort}\big(\{0\} \cup \{\operatorname{round}((1+\delta)^\ell) : \ell = 0, 1, \ldots\}\big). \tag{7}$$

The edges begin at $0, 1, 2, 3, 5, 8, \ldots$ days: recent records retain daily resolution, whereas older records are summarized over wider intervals. Each concept–bin token stores its contributing record count, most recent non-missing value, minimum and maximum non-missing values, and the timestamp of its most recent record. Consequently, the representative timestamp and the most recent non-missing value can originate from different records. The binning rule does not impose a ten-year lookback cutoff. Record eligibility and cohort construction are documented in Appendix C.4 and Appendix C.3; timestamp-availability audits are retained in Appendix F.

We then apply coverage-first selection to retain tokens within a budget of L. Let $\mathcal{B}_i$ be the binned tokens and $\mathcal{U}_i$ the set containing the most recent token of each distinct concept. Writing Recent_k for selection of up to k tokens by recency, we retain

$$\mathcal{E}_i = \begin{cases} \operatorname{Recent}_L(\mathcal{U}_i), & |\mathcal{U}_i| > L, \\ \mathcal{U}_i \cup \operatorname{Recent}_{L-|\mathcal{U}_i|}(\mathcal{B}_i \setminus \mathcal{U}_i), & |\mathcal{U}_i| \leq L. \end{cases} \tag{8}$$

This rule reserves capacity for distinct concepts before retaining repeated observations. It trades some within-concept history for broader coverage; it neither ranks concepts by clinical importance nor preserves all information. We apply the same rule in training and evaluation. Appendix E reports its effects on record length and retained history.

Each retained token is then encoded using its concept embedding and the numerical and temporal information stored in that token. For the numerical channels, we harmonize units, apply concept-specific physiological range rules, and standardize numerical values with means and standard deviations fitted on source training data. Let v_{ij}^{std} denote the most recent non-missing standardized value in token j, with within-bin extrema $v_{ij,\min}^{\text{std}}$ and $v_{ij,\max}^{\text{std}}$. The value-channel vector $\boldsymbol{v}_{ij}$ contains four active features in the main model,

$$\boldsymbol{v}_{ij} = \big[\operatorname{asinh}(v_{ij}^{\text{std}}),\ \mathbb{I}\{v_{ij}^{\text{std}} \text{ missing}\},\ \operatorname{asinh}(v_{ij,\min}^{\text{std}}),\ \operatorname{asinh}(v_{ij,\max}^{\text{std}})\big]. \tag{9}$$

Missing numerical entries are filled with zero after transformation. The implementation reserves two additional, zeroed columns for extrema-missingness flags; these do not add active input information. Consecutive-bin differences Δv^{std} and recurrence intervals Δt are alternative channels and are not enabled in this main configuration.

To encode recency, let τ_{ij} denote the elapsed days from the token's most recent contributing record to T_i. We define $u_{ij} = \log(1+\tau_{ij})/\log(1+T_0)$, where $T_0 = 3651$ is the reference timescale in days, and encode it as

$$\psi(u) = \left[\log(1+u),\ \{\sin(2\pi k u), \cos(2\pi k u)\}_{k\in\{1,2,4,8\}}\right] \in \mathbb{R}^9. \tag{10}$$

The coordinate is not clipped, so $u > 1$ is possible. The sinusoidal components span the log-time axis; they do not represent calendar periods. The first component applies a second logarithm to u, matching the implemented encoding.

The *event encoder* combines these channels with the concept vector, record count, and domain embedding. With concept c_{ij}, record count n_{ij}, and OMOP domain $\mathrm{domain}(c_{ij})$, a retained token $e_{ij} = (c_{ij}, \tau_{ij}, \boldsymbol{v}_{ij}, n_{ij})$ is embedded as

$$\boldsymbol{h}_{ij} = W_c \tilde{\boldsymbol{r}}_{c_{ij}} + W_v \boldsymbol{v}_{ij} + \boldsymbol{b}_v + W_t \psi(u_{ij}) + W_n \log(1+n_{ij}) + \boldsymbol{E}_{\mathrm{domain}}[\mathrm{domain}(c_{ij})] \in \mathbb{R}^{d_s}. \tag{11}$$

Here $\tilde{\boldsymbol{r}}_c = \boldsymbol{r}_c/\|\boldsymbol{r}_c\|_2$ is the normalized concept vector from section 3.2; the projections, value bias, and domain embeddings are learned. These components are added in a common space, with $d_s = 192$. Each event is encoded independently. The anchor aggregator described next combines these event vectors into patient-specific anchor readouts.

3.4 Semantic Anchors and Event Aggregation

Shared semantic anchors define corresponding feature columns across patients, while learned attention determines how each patient's events contribute to their contents. To construct the anchors, we center and partially whiten the concept embeddings to reduce the influence of high-variance directions, then normalize them to unit length. We denote these transformed vectors by $\check{\boldsymbol{r}}_c$. The transformation is fitted to the cached concept table (Appendix B.1); the event encoder continues to use the raw-normalized vectors $\tilde{\boldsymbol{r}}_c$.

Shared semantic anchors. We build anchors from concepts observed in a label-free source pretraining cohort whose patients are disjoint from the downstream training and evaluation partitions. Let f_c count binned occurrences of concept c in that pool and $w_c = \sqrt{f_c}$. We perform weighted Ward clustering separately within the Condition, Drug, Measurement, and Procedure domains. For clusters A and B, the merge cost is

$$D(A,B) = \frac{W_A W_B}{W_A + W_B}\|\boldsymbol{\mu}_A - \boldsymbol{\mu}_B\|_2^2, \quad W_A = \sum_{c\in A} w_c, \quad \boldsymbol{\mu}_A = \frac{1}{W_A}\sum_{c\in A} w_c \check{\boldsymbol{r}}_c. \tag{12}$$

We retain 128 clusters per domain and normalize each weighted centroid to obtain an anchor $\boldsymbol{a}_m$, giving $M = 512$ anchors in total. Square-root weights moderate the influence of frequent concepts while retaining frequency information. Cluster membership defines prototypes, not a hard assignment of patient events. The anchors remain fixed during patient-prediction training. Further construction details are provided in Appendix B.2.

We use these anchors as queries for attention pooling (Lee et al., 2019; Jaegle et al., 2021). The attention uses H heads of width $d_h = d_s/H$. For head h, the learned query and key projections give $\boldsymbol{q}_m^{(h)} = (W_q \boldsymbol{a}_m)^{(h)}$ and $\boldsymbol{k}_{ij}^{(h)} = (W_K \boldsymbol{h}_{ij})^{(h)}$. The attention score combines their learned compatibility with concept–anchor similarity:

$$s_{imj}^{(h)} = \frac{(\boldsymbol{q}_m^{(h)})^\top \boldsymbol{k}_{ij}^{(h)}}{\sqrt{d_h}} + \kappa\langle \boldsymbol{a}_m, \check{\boldsymbol{r}}_{c_{ij}}\rangle, \qquad \alpha_{imj}^{(h)} = \frac{\exp(s_{imj}^{(h)})}{\sum_{\ell\in\mathcal{E}_i\cup\{*\}}\exp(s_{im\ell}^{(h)})}. \tag{13}$$

Here κ is a fixed coefficient and $*$ is a learned null token whose similarity term is set to zero. The normalization includes all retained events and the null token, with padding masked. The similarity term depends on the concept and anchor, whereas the learned term also incorporates values, recency,

count, and domain through the event representation. Head dimensions and the similarity coefficient are specified in Table 5.

The attention weights aggregate projected event representations into the patient-specific readout for anchor m:

$$\hat{\boldsymbol{z}}_{im} = \left[F_\theta(\mathcal{E}_i)\right]^\top_{m,:} = W_o \,\mathrm{concat}_{h=1}^{H} \left[\sum_{j \in \mathcal{E}_i \cup \{*\}} \alpha_{imj}^{(h)} (W_V \boldsymbol{h}_{ij})^{(h)} \right]. \tag{14}$$

Here W_V and W_o are learned value and output projections, and $\hat{\boldsymbol{z}}_{im} \in \mathbb{R}^d$. Together with the event encoder in equation 11, this defines F_θ from retained tokens to anchor readouts. The same anchor supplies a shared aggregation query across patients, while each readout depends on that patient's observed events. Each anchor can attend to events from any clinical domain.

To form the tabular input, we subtract the mean support readout from both support and query readouts, $\boldsymbol{Z}_i = \hat{\boldsymbol{Z}}_i - \boldsymbol{\mu}_{\boldsymbol{Z}}(\mathcal{S})$. The mean is computed from support readouts alone, without labels. We then add learned anchor-column identity embeddings $\boldsymbol{E}_{\mathrm{id}} \in \mathbb{R}^{M \times d}$ and concatenate the resulting cells with encoded static covariates:

$$X_i^{\mathrm{cell}} = [\mathrm{Enc}_\phi(\boldsymbol{x}_i) \mid \boldsymbol{Z}_i + \boldsymbol{E}_{\mathrm{id}}]. \tag{15}$$

Here Enc_ϕ is the frozen scalar encoder for static covariates. The anchor readouts bypass this encoder and enter the tabular predictor directly as cell embeddings. The resulting table supports prediction as defined in equation 2.

3.5 Training and Inference

Training. We jointly train one set of parameters θ across the source clinical tasks. Each episode samples a task q and disjoint support and query patients. The frozen tabular backbone receives support labels and masks query labels; its query label-cell outputs provide predictions. We optimize the event projections, anchor aggregator, and anchor-column identity embeddings through the query loss:

$$\min_\theta \; \mathbb{E}_{q,\mathcal{S},\mathcal{Q}} \left[\frac{1}{|\mathcal{Q}|} \sum_{i \in \mathcal{Q}} \ell_q\big(g_\phi(X^{\mathrm{cell}}_{\mathcal{S}\cup\mathcal{Q}}(\theta), y_{\mathcal{S}})_i, y_i\big) \right], \qquad \phi, \boldsymbol{R}, \boldsymbol{A} \text{ fixed.} \tag{16}$$

Here ℓ_q is task-specific cross-entropy. Gradients pass through the frozen backbone to the learnable parameters θ; the backbone ϕ, concept table $\boldsymbol{R}$, and anchors $\boldsymbol{A}$ remain fixed. Training settings and initialization are specified in section 4 and Appendix B.3.

Inference. The same learned parameters are used at inference, without further updates. Internal prediction uses source support. In cross-institution prediction, labeled source instances supply the context for target-institution queries, without using target labels. Both support and query cells use the mean computed from the selected support set. Evaluation protocols, target vocabulary access, and label-use conditions are specified in section 4.1. Analyses of prediction dependence on the supplied support labels are reported in Appendix I.

4 Experiments

Datasets and implementation. We use MIMIC-IV (Johnson et al., 2023) as the source dataset and EHRSHOT (Wornow et al., 2023) for external evaluation, using structured clinical records from four OMOP domains. EXAONE Tabular (Eo et al., 2026) is the frozen tabular backbone. Data construction and model settings are detailed in Appendix C and Appendix B.3, respectively.

4.1 Prediction across institutions

We evaluate nine clinical tasks on held-out MIMIC-IV patients and corresponding EHRSHOT cohorts, with all methods sharing dataset construction and patient-level splits (Table 1). Model fitting and selection use MIMIC-IV data, and both evaluations use labeled MIMIC-IV training instances as in-context support. External evaluation permits target concept vocabulary access but uses no parameter updates or EHRSHOT labels for fitting or selection (Appendix C).

Table 1: Nine-task performance on MIMIC-IV and EHRSHOT, averaged over three seeds. ROC/PRC: AUROC/AUPRC. Avg.: task macro-average, with PHENO first averaged over its labels. **Bold**/underline: best/runner-up within each cohort, including displayed ties (Appendix C.7).

Benchmark	Model	MT		LLOS		RA		Pheno		UTI		Sepsis		PNA		MI		Fx		Avg.	
		ROC	PRC	ROC	PRC	ROC	PRC	ROC	PRC	ROC	PRC	ROC	PRC	ROC	PRC	ROC	PRC	ROC	PRC	ROC	PRC
Internal *MIMIC-IV*	BEHRT	.907	.342	.787	.619	.667	.114	.825	.411	.830	.496	.822	.303	.848	.430	.847	.188	.804	.216	.815	.347
	CDM-BERT	.937	.414	.818	.675	.742	.134	.828	.424	.866	.541	.856	.360	.870	.471	.863	.211	.841	.270	.847	.389
	BiomedBERT + BEHRT	**.970**	.640	**.863**	**.754**	.756	.145	.851	.482	.882	.586	.878	.429	.900	.608	.907	.302	.879	.436	.876	.487
	MedTok + BEHRT	.943	.566	.833	.688	.684	.089	.856	.485	.863	.542	.839	.333	.869	.521	.885	.248	.851	.336	.847	.423
	MedRep + BEHRT	.964	.639	.852	.737	.765	.181	.850	.475	.877	.578	.876	.405	.893	.565	.897	.288	.883	.435	.873	.478
	TabEHR embs. + BEHRT	.969	**.678**	.859	.741	**.782**	.161	.845	.459	.879	.565	.865	.407	.901	.607	.908	.323	.869	.412	.875	.484
	MEDS-Tab + XGBoost	.961	.644	.842	.711	.702	.170	.856	.485	.921	.679	.907	.535	.916	.675	**.955**	**.554**	.902	.403	.885	.540
	MEDS-Tab + TFM	.962	.632	.860	.752	.776	.172	.858	.504	**.926**	**.684**	.917	.564	.920	**.681**	.953	.538	.914	.411	.898	.549
	MedRep + TFM	.949	.544	.843	.718	.774	.186	.858	.490	.918	.648	.917	.584	.919	.678	.946	.469	.921	.512	.894	.537
	BiomedBERT + TFM	.957	.618	.838	.694	.734	**.203**	.863	.499	.918	.662	.913	.546	.917	.657	.949	.498	.924	.511	.890	.543
	TabEHR	.969	.647	.837	.702	.768	.168	**.864**	**.507**	.925	.682	**.925**	**.602**	**.921**	.669	.954	.520	**.926**	**.529**	**.899**	**.558**
External *EHRSHOT*	BEHRT	.722	.162	.648	.415	.573	.119	.688	.347	.769	.571	.732	.369	.739	.387	.759	.261	.653	.139	.698	.308
	CDM-BERT	.806	.247	.730	.529	.532	.106	.682	.340	.729	.440	.619	.238	.586	.235	.678	.178	.629	.122	.666	.271
	BiomedBERT + BEHRT	.900	.488	.789	.616	.582	.136	.780	.469	.791	.581	.731	.358	.743	.458	.731	.220	.812	.483	.762	.423
	MedTok + BEHRT	.815	.278	.720	.516	.594	.131	.770	.460	.779	.582	.754	.412	.781	.517	.761	.284	.776	.393	.750	.397
	MedRep + BEHRT	.847	.357	.783	.603	**.640**	**.170**	.754	.441	.773	.551	.601	.229	.694	.445	.684	.203	.802	.464	.731	.385
	TabEHR embs. + BEHRT	.858	.371	.782	.596	.605	.157	.757	.440	.804	.554	.733	.345	.753	.463	.722	.227	.794	.419	.756	.397
	MEDS-Tab + XGBoost	**.936**	.586	.795	.604	.633	.161	.778	.476	.884	.710	.729	.424	.753	.538	.642	.235	.652	.146	.756	.431
	MEDS-Tab + TFM	**.936**	**.595**	**.807**	**.637**	.581	.119	.763	.467	**.886**	**.721**	.684	.406	.743	.572	.658	.212	.652	.161	.746	.432
	MedRep + TFM	.878	.398	.785	.599	.609	.163	.778	.467	.794	.434	**.835**	**.621**	**.829**	**.643**	.757	.301	.836	.465	.789	.455
	BiomedBERT + TFM	.912	.477	.798	.614	.584	.138	.797	.492	.836	.541	.737	.450	.743	.490	.745	.322	.857	.565	.779	.454
	TabEHR	.918	.486	.797	.605	.626	.166	**.806**	**.511**	.876	.696	.828	.620	.727	.472	**.783**	**.393**	**.877**	**.643**	**.804**	**.510**

Baselines. We compare EHR sequence models and their concept-embedding variants (Li et al., 2020; Kim et al., 2025a; Gu et al., 2020; Su et al., 2025; Kim et al., 2025b), MEDS-Tab features with XGBoost or the same TFM (Oufattole et al., 2024; Chen & Guestrin, 2016), and TabEHR variants that replace its concept embeddings while retaining the aggregation and prediction architecture. Implementations and retraining procedures are specified in Appendix D.

Results. TabEHR achieves the highest average performance on both the internal MIMIC-IV benchmark and the external EHRSHOT cohort (Table 1), reaching AUROC/AUPRC of $0.899/0.558$ and $0.804/0.510$, respectively. Its advantage is more pronounced on the external cohort, where it exceeds the strongest comparator, MedRep + TFM, by 0.015 AUROC and 0.055 AUPRC. Compared with MEDS-Tab + TFM, which uses the same tabular predictor, TabEHR achieves similar internal performance but improves external AUROC and AUPRC by 0.058 and 0.078. This comparison supports learning patient features from longitudinal events for transfer beyond the source institution. The concept-embedding variants further clarify this result: MedRep and BiomedBERT embeddings within the same aggregation architecture both outperform statistical features externally, while TabEHR achieves the highest average performance. These results support the contribution of the proposed concept embeddings to the learned patient representation. The gains are not uniform across tasks; statistical baselines remain strong on mortality, length of stay, urinary tract infection, and pneumonia.

4.2 CLINICAL STRUCTURE OF THE SHARED ANCHOR SPACE

We assess whether the shared anchor space preserves clinical relationships by retrieving related concepts using their similarities to the anchors. We compare TabEHR with MedRep, MedTok, and BiomedBERT; the evaluation protocol and worked examples are provided in Appendix H.

TabEHR achieves the highest NDCG@10 across all four clinical domains, including concepts absent from source patient records (Figure 2). Retrieval uses each concept's similarities to the shared anchors, indicating that clinical relationships remain accessible through the anchor-based representation. The anchors thus provide common reference points that retain meaningful distinctions among concepts. Results for source-unobserved concepts suggest that the anchors organize clinical information beyond the source cohort's recorded vocabulary, providing shared references for aggregating related concepts across institutions. Their contribution to prediction gains remains to be established.

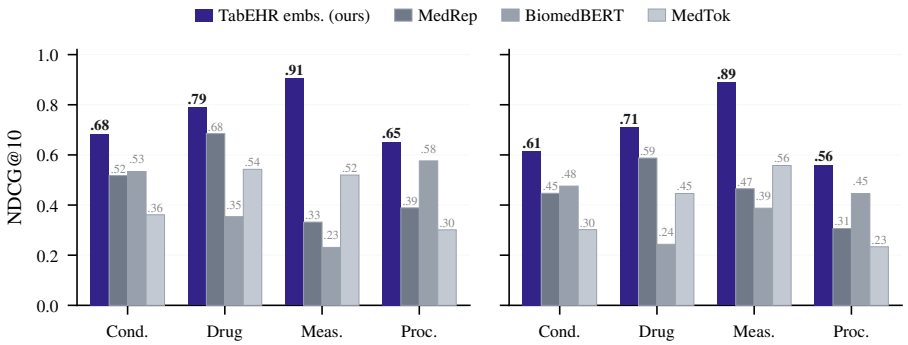

Figure 2: Clinical concept retrieval in the shared anchor space (NDCG@10; higher is better). Left: source-observed concepts. Right: source-unobserved concepts. Details are in Appendix H.2.

4.3 Few-shot prediction at an external institution

Setting. We evaluate transfer of MIMIC-IV-trained patient representations to the fifteen official EHRSHOT tasks (Wornow et al., 2023), separately from the nine-task comparison. Each binary task provides 16 labeled training examples per class, with evaluation on disjoint test patients. Source-trained representation weights remain fixed; TFM predictions use the target examples as context without parameter updates.

Table 2: Few-shot prediction on 15 EHRSHOT tasks without target-institution pretraining. Macro AUROC with 16 training examples per class.

Representation	Head	Overall (15)	Same (2)	Related (5)	Unseen (7)	CXR (14 findings)
Count	GBM	.590	.628	.567	.600	.555
	LR	.603	.624	.566	.630	.564
	TFM	.616	.641	.576	.644	.572
MedTok + BEHRT	LR	.555	.547	.548	.564	.552
	TFM	.556	.562	.543	.565	.548
MedRep + BEHRT	LR	.561	.554	.548	.573	.552
	TFM	.557	.559	.543	.567	.547
TabEHR	LR	.618	.647	.589	.637	.571
	TFM	**.659**	**.672**	**.634**	**.684**	**.581**

Comparison. We compare count features, MIMIC-trained sequence representations, and TabEHR with fitted predictors or the same frozen TFM, without target-institution pretraining (Table 2). Training-shot counts are matched, but fitted predictors additionally use target validation labels, whereas TFM rows do not. Scores average five support replicates using a single TabEHR checkpoint. Columns group tasks by overlap with source supervision; Unseen denotes outcomes not explicitly supervised at the source. CXR averages fourteen findings and contributes one task to Overall. Bold marks column maxima. Task definitions, label budgets, checkpoints, and extended comparisons are detailed in Appendix K.

Results. TabEHR with the TFM achieves the highest macro AUROC among these methods over all fifteen tasks and within each reported group (Table 2). On the seven outcomes not explicitly supervised by the source tasks, it improves AUROC from .644 with count features and the same TFM to .684; the overall mean increases from .616 to .659. These results support reusing source-supervised patient representations for new clinical prediction tasks with limited target labels. The gains describe the learned representation–predictor combination, including its source supervision, rather than the effect of semantic anchors alone.

5 Discussion and Limitations

TabEHR's larger gains over statistical features with the same TFM in external evaluation highlight the importance of patient representation for transfer. Concept semantics organize both events and shared feature columns, and retrieval analysis shows that clinical relationships are preserved even for concepts absent from source records. Their contribution to prediction gains remains to be isolated.

Evaluation is limited to one external institution and one TFM backbone, EXAONE Tabular. Broader generalization across healthcare settings, prediction tasks, and tabular backbones requires further study.

REFERENCES

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.

Edward Choi, Mohammad Taha Bahadori, Joshua A. Kulas, Andy Schuetz, Walter F. Stewart, and Jimeng Sun. RETAIN: An interpretable predictive model for healthcare using reverse time attention mechanism. In *Advances in Neural Information Processing Systems*, volume 29, pp. 3504–3512, 2016a. URL https://proceedings.neurips.cc/paper/2016/hash/231141b34c82aa95e48810a9d1b33a79-Abstract.html.

Edward Choi, Mohammad Taha Bahadori, Andy Schuetz, Walter F. Stewart, and Jimeng Sun. Doctor AI: Predicting clinical events via recurrent neural networks. In *Proceedings of the 1st Machine Learning for Healthcare Conference*, volume 56 of *Proceedings of Machine Learning Research*, pp. 301–318, 2016b. URL https://proceedings.mlr.press/v56/Choi16.html.

Edward Choi, Mohammad Taha Bahadori, Le Song, Walter F. Stewart, and Jimeng Sun. GRAM: Graph-based attention model for healthcare representation learning. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 787–795, 2017. URL https://arxiv.org/abs/1611.07012.

Yilang Ding, Jiawen Ren, Jiaying Lu, Gloria Hyunjung Kwak, Armin Iraji, and Alex Fedorov. Longitudinal progression prediction of Alzheimer's disease with tabular foundation model. *arXiv preprint arXiv:2508.17649*, 2025. URL https://arxiv.org/abs/2508.17649.

Jun Ni Du, L. Adamek, M. Kryukov, F. Dormont, Z. Bar-Joseph, S. Jager, and B. Rufino. Explainable transformer models for clinical prediction tasks on structured electronic health records. *arXiv preprint arXiv:2608.20315*, 2026. BERT-LER.

Moonjung Eo, Min-Kook Suh, Hye-Seung Cho, Jiwon Kim, Seoyoon Kim, Sangjun Nam, and Soonyoung Lee. EXAONE Tabular 1.0: Technical report. *arXiv preprint arXiv:2608.25774*, 2026. URL https://arxiv.org/abs/2608.25774.

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 *Proceedings of the 4th Machine Learning for Health Symposium*, volume 259 of *Proceedings of Machine Learning Research*, pp. 291–307, 2025. URL https://proceedings.mlr.press/v259/fallahpour25a.html.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing, 2020. URL https://arxiv.org/abs/2007.15779.

Hrayr Harutyunyan, Hrant Khachatrian, David C. Kale, Greg Ver Steeg, and Aram Galstyan. Multitask learning and benchmarking with clinical time series data. *Scientific Data*, 6(1):96, 2019. doi: 10.1038/s41597-019-0103-9.

Healthcare Cost and Utilization Project (HCUP). Clinical classifications software (CCS) for ICD-9-CM and ICD-10-CM. Agency for Healthcare Research and Quality, Rockville, MD. https://hcup-us.ahrq.gov/toolssoftware/ccs/ccs.jsp.

George Hripcsak, Jon D. Duke, Nigam H. Shah, Christian G. Robinson, et al. Observational health data sciences and informatics (OHDSI): Opportunities for observational researchers. In *Studies in Health Technology and Informatics*, volume 216, pp. 574–578, 2015.

Kyunghoon Hur, Jiyoung Lee, Jungwoo Oh, Wesley Price, Younghak Kim, and Edward Choi. Unifying heterogeneous electronic health records systems via text-based code embedding. In *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pp. 183–203, 2022. URL https://proceedings.mlr.press/v174/hur22a.html.

Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 4651–4664, 2021. URL https://proceedings.mlr.press/v139/jaegle21a.html.

Benjamin Jäger, Nick Erickson, Léo Grinsztajn, Felix Birkel, Klemens Flöge, Oscar Key, Kürşat Kaya, Jonas Kübler, Adèle Frankel, Tobias Schröder, Anurag Garg, Jan Hendrik Metzen, David Salinas, Simon Bing, Kristina Collins, Tuana Çelik, Vahid Balazadeh, Lydia Sidhoum, Tomás Pereda, Brendan Roof, Andrej Tschalzev, Siyuan Guo, Philipp Singer, Lennart Purucker, Jake Robertson, Marie Salmon, Philipp Jund, Jerry Chen, Diana Kriuchkova, Arthur Cahu, Eliott Kalfon, Adrian Hayler, Georg Grab, Vitor Monteiro, Lilly Wehrhahn, Dominik Safaric, Clara Cornu, Alan Arazi, Rylee Grace, Simone Alessi, Mihir Manium, Bernhard Schölkopf, Yann LeCun, Madelon Hulsebos, Sauraj Gambhir, Noah Hollmann, and Frank Hutter. TabPFN-3.5: Technical report. *arXiv preprint arXiv:2609.17895*, 2026. URL https://arxiv.org/abs/2609.17895.

Pengcheng Jiang, Cao Xiao, Adam Cross, and Jimeng Sun. GraphCare: Enhancing healthcare predictions with personalized knowledge graphs. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/484d254ff80e99d543159440a06db0de-Abstract-Conference.html.

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, 2023.

J. Kim, J. S. Kim, J.-H. Lee, et al. Pretrained patient trajectories for adverse drug event prediction using common data model-based electronic health records. *Communications Medicine*, 5:232, 2025a. CDM-BERT.

Junmo Kim, Namkyeong Lee, Jiwon Kim, and Kwangsoo Kim. MedRep: Medical concept representations for general electronic health record foundation models. *arXiv preprint arXiv:2504.08329*, 2025b.

Thomas A. Lasko, Joshua C. Denny, and Mia A. Levy. Computational phenotype discovery using unsupervised feature learning over noisy, sparse, and irregular clinical data. *PLOS ONE*, 8(6): e66341, 2013. doi: 10.1371/journal.pone.0066341. URL https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0066341.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set Transformer: A framework for attention-based permutation-invariant neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3744–3753, 2019. URL https://proceedings.mlr.press/v97/lee19d.html.

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

Yikuan Li, Mohammad Mamouei, Gholamreza Salimi-Khorshidi, Shishir Rao, Abdelaali Hassaine, Dexter Canoy, Thomas Lukasiewicz, and Kazem Rahimi. Hi-BEHRT: Hierarchical transformer-based model for accurate prediction of clinical events using multimodal longitudinal electronic health records. *IEEE Journal of Biomedical and Health Informatics*, 27(2):1106–1117, 2023. doi: 10.1109/JBHI.2022.3224727.

Nassim Oufattole, Teya Bergamaschi, Aleksia Kolo, Hyewon Jeong, Hanna Gaggin, Collin M. Stultz, and Matthew B. A. McDermott. MEDS-Tab: Automated tabularization and baseline methods for MEDS datasets. *arXiv preprint arXiv:2411.00200*, 2024.

Minh-Khoi Pham, Luca Cotugno, Alina Sîrbu, Tai Tan Mai, Martin Crane, and Marija Bezbradica. Tabular foundation models for clinical survival analysis via survival-aware adaptation. *arXiv preprint arXiv:2606.12006*, 2026a. URL https://arxiv.org/abs/2606.12006.

Minh-Khoi Pham, Thang-Long Nguyen Ho, Thao Thi Phuong Dao, Tai Tan Mai, Minh-Triet Tran, Marie E. Ward, Una Geary, Rob Brennan, Nick McDonald, Martin Crane, and Marija Bezbradica. Retrieval-aligned tabular foundation models enable robust clinical risk prediction in electronic health records under clinically realistic constraints. *arXiv preprint arXiv:2604.01841*, 2026b. URL https://arxiv.org/abs/2604.01841v4. Version 4.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICLv2: A better, faster, scalable, and open tabular foundation model. *arXiv preprint arXiv:2602.11139*, 2026. URL https://arxiv.org/abs/2602.11139.

Alvin Rajkomar, Eyal Oren, Kai Chen, Andrew M. Dai, Nissan Hajaj, Michaela Hardt, Peter J. Liu, Xiaobing Liu, Jake Marcus, Mimi Sun, Patrik Sundberg, Hector Yee, Kun Zhang, Yi Zhang, Gerardo Flores, Gavin E. Duggan, Jamie Irvine, Quoc Le, Kurt Litsch, Alexander Mossin, Justin Tansuwan, De Wang, James Wexler, Jimbo Wilson, Dana Ludwig, Samuel L. Volchenboum, Katherine Chou, Michael Pearson, Srinivasan Madabushi, Nigam H. Shah, Atul J. Butte, Michael D. Howell, Claire Cui, Greg S. Corrado, and Jeffrey Dean. Scalable and accurate deep learning with electronic health records. *npj Digital Medicine*, 1:18, 2018. doi: 10.1038/s41746-018-0029-1. URL https://www.nature.com/articles/s41746-018-0029-1.

Laila Rasmy, Yang Xiang, Ziqian Xie, Cui Tao, and Degui Zhi. Med-BERT: Pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction. *npj Digital Medicine*, 4(1):86, 2021. doi: 10.1038/s41746-021-00455-y. URL https://www.nature.com/articles/s41746-021-00455-y.

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. doi: 10.1016/j.jbi.2020.103637. URL https://doi.org/10.1016/j.jbi.2020.103637.

Xiaorui Su, Shvat Messica, Yepeng Huang, Ruth Johnson, Lukas Fesser, Shanghua Gao, Faryad Sahneh, and Marinka Zitnik. Multimodal medical code tokenizer. In *International Conference on Machine Learning*, 2025. arXiv:2502.04397.

Michael Wornow, Rahul Thapa, Ethan Steinberg, Jason A. Fries, and Nigam H. Shah. EHRSHOT: An EHR benchmark for few-shot evaluation of foundation models. In *Advances in Neural Information Processing Systems (Datasets and Benchmarks Track)*, 2023.

Michael Wornow, Suhana Bedi, Miguel Angel Fuentes Hernandez, Ethan Steinberg, Jason Alan Fries, Christopher Ré, Sanmi Koyejo, and Nigam H. Shah. Context clues: Evaluating long context models for clinical prediction tasks on EHRs. In *International Conference on Learning Representations*, 2025. arXiv:2412.16178.

A NOTATION AND CONFIGURATION

A.1 NOTATION AND DIMENSIONS

Instances and time. A row is a prediction instance: a patient together with prediction time T_i. A patient can contribute multiple instances. A retained event token is $e_{ij} = (c_{ij}, \tau_{ij}, \boldsymbol{v}_{ij}, n_{ij})$, where τ_{ij} is elapsed time in days, $u_{ij} = \log(1+\tau_{ij})/\log(1+T_0)$ is the normalized log-time coordinate, and $T_0 = 3651$ days is a scale, not a maximum history. The concept determines the domain $\mathrm{domain}(c_{ij})$. The standardized numerical value v_{ij}^{std} and within-bin extrema $v_{ij,\min}^{\mathrm{std}}$ and $v_{ij,\max}^{\mathrm{std}}$ form the value-channel vector $\boldsymbol{v}_{ij}$ after asinh transformation and the addition of a missingness indicator. These numerical inputs are distinct from the anchor readouts $\hat{\boldsymbol{z}}_{im}$ and their matrix $\hat{\boldsymbol{Z}}_i$.

Representations and episode dependence. $\boldsymbol{r}_c$ is a cached concept embedding; $\tilde{\boldsymbol{r}}_c$ is its raw L_2-normalized form; $\check{\boldsymbol{r}}_c$ is its centered, partially whitened, normalized form. The map F_θ comprises event encoding and anchor aggregation, producing $\hat{\boldsymbol{Z}}_i$ from the patient's retained event tokens without using other patients or labels. It ends at the anchor readouts, before support centering and cell integration. Support centering produces $\boldsymbol{Z}_i = \hat{\boldsymbol{Z}}_i - \boldsymbol{\mu}_{\boldsymbol{Z}}(\mathcal{S})$; these centered readouts depend on the episode. The final anchor-cell embeddings are $\boldsymbol{Z}_i + \boldsymbol{E}_{\mathrm{id}}$. g_ϕ consumes the assembled episode and support labels. Its frozen static input path can also use support-derived normalization.

Reserved symbols. $\mathbb{R}$ denotes the real numbers and σ the logistic sigmoid; value squashing is written asinh. Vectors and matrices are bold italic. The embedding table $\boldsymbol{R}$ is distinct from $\mathbb{R}$. $\varnothing$ denotes a missing entry, while $*$ denotes the attention null token. $\alpha_{imj}^{(h)}$ is an attention weight; α_{w} is the partial-whitening exponent. The main model has no separate scalar multiplier at the cell-input junction.

Table 3: Dimensions of the main configuration.

Symbol	Meaning	Value
d_r	Cached concept dimension	768
d	Backbone cell width	192
d_s	Summarizer internal width	192
M	Number of anchors / anchor columns	512
H, d_h	Attention heads / head width	4 / 48
d_x	Static covariates (age and sex)	2
L	Event-token budget per instance	512
C	Active / physical value-input dimensions	4 / 6
d_ψ	Recency-encoding dimension	9

A.2 SYMBOL TABLE

Table 4 collects the notation used across the five Methods subsections.

Table 4: Symbols used in the Methods.

Symbol	Meaning
$\mathcal{C}, \mathcal{T}$	Concept vocabulary and clinical task set
$q, \mathcal{Y}_q, y_i$	Clinical task, its label space, and instance target
i, j, m, h	Prediction instance, event, anchor, and attention head
$\mathcal{S}, \mathcal{Q}; n_{\mathcal{S}}, n_{\mathcal{Q}}$	Support and query sets; their sizes
$\mathcal{H}_i$	Clinical records available before the prediction time
$\mathcal{E}_i, N_i$	Retained event set and its size
T_i, τ_{ij}	Prediction timestamp; elapsed days before prediction
u_{ij}, T_0	Normalized log-time coordinate; its reference scale
$\boldsymbol{x}_i$	Static covariates
v_{ij}^{std}	Standardized numerical value before asinh transformation
$v_{ij,\min}^{\mathrm{std}}, v_{ij,\max}^{\mathrm{std}}$	Standardized within-bin minimum and maximum
$\boldsymbol{v}_{ij}, n_{ij}$	Value-channel vector and contributing record count
$\boldsymbol{h}_{ij}$	Event representation after channel projection and addition
b_k, δ	Temporal bin edges and geometric growth parameter
$\boldsymbol{r}_c, \boldsymbol{R}$	Cached concept embedding and complete table
$\tilde{\boldsymbol{r}}_c, \check{\boldsymbol{r}}_c$	Raw-normalized and whitened-normalized concept vectors
$\boldsymbol{a}_m, \boldsymbol{A}$	Fixed unit-norm anchor and anchor matrix
f_c, w_c	Binned source-pool frequency and its square root
κ	Coefficient of the concept–anchor similarity term
$s_{imj}^{(h)}, \alpha_{imj}^{(h)}$	Attention score and normalized weight
F_θ	Composition of event encoding and anchor aggregation, mapping retained tokens to anchor readouts
θ	Trainable prediction-input parameters: event encoder, anchor aggregator, and column identities
$\hat{\boldsymbol{z}}_{im}, \hat{\boldsymbol{Z}}_i$	One anchor readout and the matrix of all readouts, before support centering
$\boldsymbol{\mu}_{\boldsymbol{Z}}(\mathcal{S}), \boldsymbol{Z}_i$	Support mean and centered readout
$\boldsymbol{E}_{\mathrm{id}}$	Learned slot-identity matrix, added after centering
X_i^{cell}	Assembled static and anchor-cell embeddings supplied to the backbone
g_ϕ, ϕ	Frozen tabular predictor and its parameters
Enc_ϕ	Frozen scalar encoder for static covariates
η	Concept text-encoder parameters, trained before patient prediction
$\mathcal{D}^{\mathrm{src}}, \mathcal{D}^{\mathrm{tgt}}$	Source and target institutions
Ω	Artifacts fixed before patient-level evaluation
K	Number of retained statistical-baseline features

A.3 MAIN CONFIGURATION

Table 5 specifies the main-table configuration. Other anchor, channel, and protocol variants are identified separately. The saved configuration and final checkpoint linkage are tracked under M1; this is distinct from the concept-export provenance checks CE1–CE4.

Table 5: Main TabEHR configuration. Optimization details appear in Table 6.

Component	Setting
Anchor set	$\sqrt{f}$-weighted Ward; 128 anchors per domain, $M = 512$
Routing geometry	Partial whitening, $\alpha_{\mathrm{w}} = 0.5$, $\epsilon = 10^{-4}$
Similarity coefficient	Fixed $\kappa = 7.9815$
Value-channel vector	$\mathrm{asinh}(v^{\mathrm{std}})$, value-missingness indicator, $\mathrm{asinh}(v^{\mathrm{std}}_{\mathrm{min}})$, $\mathrm{asinh}(v^{\mathrm{std}}_{\mathrm{max}})$
Temporal grid	Rounded geometric edges, $\delta = 0.5$
Time encoding	Nine features on normalized log-time; $T_0 = 3651$ days
Selection	Coverage-first, $L = 512$
Summarizer	Cross-attention, $d_s = 192$, $H = 4$, $d_h = 48$
Centering and identity	Support mean only; slot identity added afterward
Predictor	Frozen tabular foundation model, $d = 192$
Episode	$n_{\mathcal{S}} \in [256, 1024]$, $n_{\mathcal{Q}} = 256$
Prediction training	Aggregator and slot identities learned; backbone, concept table, and anchors fixed

B METHOD AND IMPLEMENTATION DETAILS

B.1 CONCEPT DESCRIPTION AND EMBEDDING CONSTRUCTION

Ontology-informed description generation. For each OMOP concept in the synthesis corpus, we construct a structured generation input from its concept name, domain, vocabulary, and typed one-hop neighborhood in the OMOP vocabulary. Relation types are preserved rather than flattened into an unordered list, allowing the generator to distinguish hierarchical relations from anatomical and other domain-specific relations.

A shared system prompt defines four complementary clinical roles—an internist, pharmacist, clinical scientist, and informaticist—together with the criteria for evaluating ontology relations and the rules for generating clinical descriptions. Given the concept-specific input, the language model first assigns each supplied ontology edge one of four verdicts: VALID, OUTDATED, INCORRECT, or INSUFFICIENT, with a short natural-language justification. It then generates one description from each of the four clinical perspectives and integrates them into a single clinical synthesis.

The synthesis is instructed to preserve clinically relevant ontology evidence while avoiding the propagation of relations judged outdated or incorrect. This filtering is performed automatically by the language model and does not constitute clinician validation. The role-specific descriptions target 60–90 words each, and the integrated synthesis targets 220–300 words, with a broader accepted generation range of 150–400 words. Concepts are referred to by their natural-language names in the generated text rather than by concept-ID tokens. The resulting corpus contains 493,560 concepts with usable integrated syntheses.

Illustrative generation example. As an example, consider OMOP concept 140609, *Partial thickness burn of multiple sites of trunk*, a SNOMED concept in the Condition domain. Its structured one-hop context contains the following ontology relations:

> **Hierarchy:**
> *Is a* → Partial thickness burn of trunk
> *Is a* → Partial thickness burn of multiple sites
>
> **Anatomy:**
> *Has finding site* → Structure of multiple topographic sites
> *Has finding site* → Skin structure of trunk

For this concept, all four supplied edges are classified as VALID, each with a short justification. For example, the relation to *Partial thickness burn of trunk* is supported because the seed concept describes a more specific instance involving multiple sites of the trunk.

The four role-specific views describe the same concept from complementary clinical perspectives. For example, the internist view emphasizes clinical presentation and management:

> *Partial thickness burns of multiple sites of the trunk present with pain, redness, and blistering. These burns affect the epidermis and part of the dermis, leading to potential complications such as infection.*

The final synthesis integrates information from the role-specific views with the validated ontology neighborhood. An excerpt is:

> *Partial thickness burns of multiple sites of the trunk are characterized by damage to the epidermis and part of the dermis, leading to symptoms such as pain, redness, and blistering. . . . The anatomical relevance is highlighted by its association with the skin structure of the trunk and multiple topographic sites.*

This example illustrates how the generated document enriches the seed concept with typed relational structure and complementary clinical interpretations, rather than representing the concept from its name alone.

Concept-level representation learning. We train a concept encoder H_η on the generated clinical descriptions. Training first uses masked-language modeling (MLM) and subsequently combines

MLM with a concept-level contrastive objective. Input sequences are truncated to 384 tokens, and MLM masks tokens with probability 0.15. Both integrated syntheses and role-specific descriptions are used as natural-language training documents.

For contrastive learning, let x_c^{syn} denote the integrated synthesis of concept c, and let $x_c^{(k)}$ denote one of its four role-specific descriptions. One role-specific description is sampled uniformly for each concept during training. For a text sequence x, we obtain its representation by mean-pooling the final hidden states over non-padding tokens:

$$g_\eta(x) = \mathrm{MeanPool}\left(H_\eta(x)\right). \tag{17}$$

The synthesis and role-specific representations are then ℓ_2-normalized:

$$\mathbf{s}_c = \frac{g_\eta(x_c^{\mathrm{syn}})}{\|g_\eta(x_c^{\mathrm{syn}})\|_2}, \qquad \mathbf{v}_c = \frac{g_\eta(x_c^{(k)})}{\left\|g_\eta(x_c^{(k)})\right\|_2}. \tag{18}$$

For a minibatch of B concepts, let $\mathbf{S} \in \mathbb{R}^{B\times d_r}$ and $\mathbf{V} \in \mathbb{R}^{B\times d_r}$ stack the normalized synthesis and role-specific representations, respectively. The synthesis and sampled role-specific description of the same concept form a positive pair, while representations of other concepts in the minibatch provide negatives. We use the symmetric InfoNCE objective

$$\mathcal{L}_{\mathrm{con}} = \frac{1}{2}\left[\mathrm{CE}\left(\frac{\mathbf{S}\mathbf{V}^\top}{\tau}, \mathbf{y}\right) + \mathrm{CE}\left(\frac{\mathbf{V}\mathbf{S}^\top}{\tau}, \mathbf{y}\right)\right], \tag{19}$$

where $\mathbf{y} = (1, \ldots, B)$ identifies the matching concepts and $\tau = 0.05$ is the contrastive temperature.

During joint training, MLM is applied to both members of each synthesis–perspective pair. The resulting objective is

$$\mathcal{L}_{\mathrm{concept}} = \mathcal{L}_{\mathrm{MLM}} + \lambda_{\mathrm{con}}\mathcal{L}_{\mathrm{con}}, \qquad \lambda_{\mathrm{con}} = 0.5. \tag{20}$$

We optimize the encoder with AdamW using weight decay 0.01 and a linear learning-rate schedule with 10% warm-up. Five percent of the concept corpus is reserved for validation.

Cached concept representations. After concept-level training, we construct the fixed concept representation table used by TabEHR. Each concept is represented by encoding its integrated synthesis and mean-pooling the final hidden states over non-padding tokens:

$$\boldsymbol{r}_c = \mathrm{MeanPool}\left(H_\eta(x_c^{\mathrm{syn}})\right) \in \mathbb{R}^{768}. \tag{21}$$

The cached concept vector is the mean-pooled encoder output itself; the ℓ_2 normalization above is used for the contrastive objective and is not part of the exported base representation. The concept representation table remains fixed throughout patient-prediction training.

For numerical measurements, we additionally construct measurement-specific representations that distinguish value ranges. For a measurement assigned to the k-th decile, a short natural-language statement indicating its ordinal range is appended to the concept description, and the resulting text is encoded separately. This allows different ranges of the same measurement to receive distinct semantic representations.

The cached concept vectors subsequently serve two roles in TabEHR. First, they provide the semantic representation of clinical concepts when constructing patient-event representations. Second, their geometry provides the basis for the fixed semantic anchors described in section 3.4. Centering, spectral transformation, and normalization used specifically for anchor construction are applied downstream to this base representation space and are described separately with the semantic-anchor construction.

Relational analyses of the frozen concept embeddings are reported in Appendix H.1.

B.2 Anchor construction and readout semantics

Construction pool. The W2All source pool and candidate-vocabulary manifest are specified in Appendix C.2. Frequency f_c counts binned tokens in that pool, not individual patients or all raw

records represented by a token. Counts can depend on which patient windows enter the pool. A target concept absent from the construction pool can still contribute through its cached embedding and attention under the vocabulary-access policy in Appendix C.5.

Within each domain, singleton concept clusters are merged using equation 12 until 128 remain. Final prototypes are the normalized weighted means of their member vectors. The saved anchors are reused across prediction runs. The available construction record does not fully specify repeated-window accounting or merge tie handling, so it does not establish bitwise reconstruction of the anchor artifact.

What a slot represents. The anchor $\boldsymbol{a}_m$ is fixed, the uncentered readout $\hat{\boldsymbol{z}}_{im}$ is patient-specific, and the centered cell also depends on the support cohort. These are distinct objects. The attention mask permits all valid events regardless of the anchor's construction domain. Thus a Condition-derived anchor can read Drug or Measurement events; its actual domain composition must be measured before attaching a clinical interpretation. An anchor need not correspond to a single diagnosis, and its readout is a vector representation rather than a disease probability.

The null token is learned in the event-token space and passes through the same key and value projections as real events, with zero concept-prior bias. A diagnostic activity score can be defined as

$$a_{im}^{\text{event}} = 1 - \frac{1}{H}\sum_{h=1}^{H} \alpha_{im*}^{(h)}. \tag{22}$$

It measures attention allocated to real events, not disease probability. Because the softmax is over a patient's retained events, surrounding records change normalized weights. The model does not contextualize event content before pooling or impose a sequence order. Reordering tokens together with their attributes is an invariance check; changing timestamps is a different intervention that can change both keys and values. Interactions among the resulting cells occur in the tabular backbone.

Interpretation of temporal examples. Any visualization at a historical prediction time must be recomputed from the corresponding record prefix, including its binning and recency coordinates. The support set must be held fixed when comparing centered cells. A response to related concepts in the absence of a particular diagnosis code is a response of a representation, not evidence that an undocumented diagnosis was detected. Anchor-level domain mass, null weight, and attended-event composition should accompany such examples.

B.3 Patient-prediction training

Backbone and episode configuration. EXAONE Tabular's reported pretraining uses synthetic tabular tasks generated from a structural causal model (SCM) prior. We reuse its pretrained classification checkpoint; the exact artifact is tracked under B1. The main scratch run learns from source `ft_train` episodes. The pretraining cohort supplies anchor construction statistics; it is not the supervised episode pool for this run. The tabular backbone specified in section 4, including its scalar encoder and output decoder, the concept table, and the anchors are frozen. The output projection and slot identities start at zero; the null event token is also initialized to zero and then learned. These zero-valued added cells do not establish equivalence to a backbone with those columns removed.

One aggregator is trained across the nine source tasks. Each episode samples a task and disjoint support and query rows; a phenotyping episode uses one binary phenotype label. The main input retains up to $L = 512$ binned events by coverage-first selection, with last/min/max value channels and no Δv^{std} or Δt. Support centering and post-centering slot identities are applied throughout. Table 6 records the complete training recipe. Hardware-dependent microbatch arrangements preserve an effective batch of 16 episodes per optimizer step.

Cell integration details. Slot identities are added after support centering so that subtracting the mean does not cancel them. Anchor readouts enter at the backbone's cell width, without an intervening scalar projection or independent slot-scale parameter. With two static covariates, 512 anchor cells, and the label cell, each row contains 515 columns. Support label cells hold the selected task's labels; query label cells are masked and their outputs are decoded for prediction.

Table 6: Main TabEHR source-training configuration. The scheduled training budget and the checkpoint selected for reporting are distinct. Selected checkpoints are reported below.

Setting	Value
Initialization	Scratch aggregator; pretrained frozen backbone
Updated parameters	508,800 in the main patient-prediction model
Event / output width	192 / 192; 4 anchor-attention heads
Support / query size	256–1024 / 256 per episode
Task sampling	Proportional to eligible task size; one binary phenotype label per phenotyping episode
Backbone	TFM (section 4); frozen scalar encoder, transformer, and decoder
Effective batch	16 episodes per optimizer step
Recorded accumulation	1 episode/device, 2 devices, 8 accumulation steps
Optimizers	Muon for hidden key, value, and output matrices; AdamW for input/query projections and other aggregator parameters
Learning rates	3×10^{-4} for both aggregator groups
Weight decay	0.1 for the Muon group; 0 for the aggregator input group
Schedule	1,000 warmup steps; stable learning rate through step 4,000; cosine decay to step 12,000
Precision and EMA	BF16 mixed precision; EMA decay 0.9999
Checkpoint selection	Highest source-validation macro AUROC

History augmentation. The training loader can shorten an instance's available history after coverage-first token selection. In the main configuration it keeps full history with probability 0.5; otherwise it samples a horizon H_i with $\log H_i \sim \mathcal{U}(\log 7, \log 3651)$ and masks tokens older than that horizon. This training augmentation is distinct from temporal binning and the recency normalization scale. Evaluation does not apply random history augmentation. Masked tokens are not replaced by refilling the budget.

Configuration provenance. The main scratch run uses the `ch4lrsqdc` configuration. Its saved model configuration has backbone width 192 and no internal-width override, giving summarizer width 192 and per-head width 48. It enables within-bin extrema, disables Δv^{std} and Δt, masks extrema-missingness columns, and adds slot identities after support centering. The value projection has six physical inputs with layout $[\mathrm{asinh}(v^{\mathrm{std}}), \mathbb{I}\{v^{\mathrm{std}} \text{ missing}\}, \mathrm{asinh}(v^{\mathrm{std}}_{\min}), 0, \mathrm{asinh}(v^{\mathrm{std}}_{\max}), 0]$. The four-feature equation in the main text is its effective active-input form. Saved configurations and source-validation selection curves have been archived for the three main seeds. The complete link from each evaluation invocation to its weight export (including EMA choice) and precision settings has not been verified. The configuration below distinguishes saved training settings from this evaluator provenance.

Run accounting. The nine-task comparison uses the common seed set $\{100, 200, 300\}$ and reports the mean of seed-specific metrics. Sequence-model checkpoints are selected by task-specific MIMIC-IV validation AUROC. TabEHR and the concept-embedding variants with its frozen tabular backbone select one checkpoint per seed using the macro AUROC across the nine source-validation tasks. The same selected checkpoint is evaluated across all nine tasks, internally and externally.

The current results record TabEHR checkpoints at steps 6,000, 7,000, and 7,000 for seeds 100, 200, and 300, respectively. The corresponding BiomedBERT + TFM checkpoints are at steps 5,000, 7,000, and 6,000. The scheduled training budget is 12,000 steps; it is distinct from the selected checkpoint. Between-seed variation is reported in Appendix J; patient-paired uncertainty is not reported.

The in-context predictors use each task's complete eligible MIMIC `ft_train` split as support for internal and external prediction (section 4.1). The pretrained tabular backbone itself is not retrained for different seeds. Seeds apply to stochastic training and input construction where applicable; repeated deterministic inference with identical inputs and support is not an independent training run. The MedRep + TFM aggregators are still training, so that row of Table 1 is selected from the steps available so far rather than from a completed run.

Scope of transfer. Support origins, target-label use, and vocabulary access are defined together in Appendix C.5. Support centering removes the support mean in slot space; it does not guarantee removal of arbitrary cross-institution distribution shifts.

C Datasets and Nine-Task Evaluation

This appendix specifies the data sources, cohort construction, patient partitions, and data-access rules for the nine-task comparison in section 4.1. Temporal input audits and baseline implementations are detailed in Appendix F and Appendix D.

C.1 Data sources and clinical records

MIMIC-IV (Johnson et al., 2023), from Beth Israel Deaconess Medical Center, is the source institution. EHRSHOT (Wornow et al., 2023), from Stanford Medicine, is the external institution. Structured records are mapped to the OMOP common data model (Hripcsak et al., 2015). We use conditions, drugs, measurements, and procedures, retaining concept identifiers, recorded timestamps, and numerical values where available. An event's domain is inherited from its source OMOP table; the concept-to-domain audit is in Appendix F.1. Age and sex are the two static covariates in the main tabular comparisons.

The institutions differ in record length and task-specific positive rates, as documented in Appendix E and Table 7.

The nine-task datasets are reconstructed under common cohort eligibility, label definitions, prediction cutoffs, and patient partitions for all methods. Method-specific processing is documented in Appendix D; sharing a data construction does not imply identical tokenization, numerical processing, or concept coverage. The exact MIMIC-IV release, EHRSHOT extraction, OMOP vocabulary snapshot, and observation periods of the materialized inputs remain to be linked to their conversion manifests.

C.2 Source partitions and construction pools

The source cohort construction reserves $258{,}632$ of $304{,}274$ eligible patients for EHR pretraining and $45{,}642$ for downstream cohort construction. These are partition-level counts before the task-specific filtering below. The sequence baselines use this reserved pretraining partition; it is distinct from the smaller materialized pool used to construct TabEHR's anchors.

Source patients reserved for EHR pretraining are disjoint from the downstream patients used for supervised training, validation, and testing. The materialized label-free pool used to construct anchors contains $175{,}412$ patients; its intersection with each downstream split is zero. This pool defines concept frequencies and the anchor construction universe. The main TabEHR aggregator is instead trained from scratch on labeled downstream source training episodes, with the concept vectors and anchors held fixed.

The frozen downstream assignment contains $15{,}886$ training, $5{,}295$ validation, and $5{,}294$ test patients, approximately a $60/20/20$ split. After task eligibility and input filtering, the current nine-task datasets retain $15{,}683$, $5{,}227$, and $5{,}226$ distinct patients in these splits, respectively ($26{,}136$ in total). A patient can contribute to several tasks but retains the same split in all of them. Model seeds do not redraw this assignment. The corresponding external datasets contain $4{,}444$ distinct eligible EHRSHOT patients across tasks. These are evaluation-specific cohorts, not the sizes of the complete released databases.

Anchor construction pool. The W2All construction uses the label-free source pool `tabehr_pool_v4Pcov`. Its reported manifest contains $175{,}412$ patients, $23{,}832$ observed concepts, and $40{,}180{,}451$ binned token occurrences. The source-observed concepts form the candidate anchor vocabulary. This vocabulary is distinct from the cached embedding table, which also contains target concepts (Appendix C.5). These corpus counts describe anchor construction, not the supervised training examples for the main aggregator.

C.3 Nine-task cohorts and label construction

Following the task families of MedRep (Kim et al., 2025b), the nine tasks are in-hospital mortality (MT), a hospital stay of at least seven days (LLOS), 30-day readmission (RA), urinary tract infection (UTI), sepsis, pneumonia (PNA), myocardial infarction (MI), closed fracture (FX), and

24-label phenotyping (PHENO). We revise the label construction for MI and PHENO. The main comparison's MI task excludes eight concepts denoting prior myocardial infarction, as described in Appendix C.3.1. PHENO labels are reconstructed from source diagnosis codes using CCS mappings, as detailed below. Remaining task-level eligibility, prediction-time, and labeler settings must remain tied to the final materialized inputs. The exclusion is present in the labels the main comparison consumes, not only in the labeler specification: the materialized MI label file matches the archived prior-infarction-excluded variant instance for instance and positive for positive (26,242 instances with 1,458 positives at the source, 4,022 with 408 at the target), whereas the unrevised labels hold 2,818 and 694 positives over near-identical instance counts.

Table 7 summarizes the task-specific datasets used in the main comparison. The partition sizes above count distinct patients across tasks; the table reports prediction instances separately for each task, so its rows should not be summed to obtain unique-patient counts. The task families follow Kim et al. (2025b); the current cohorts use our shared patient-level assignment rather than independent task-level splits.

Table 7: Nine-task cohort sizes and evaluation positive rates. Counts denote prediction instances; PHENO prevalence is averaged over its 24 labels (Table 8).

	MIMIC-IV instances			EHRSHOT	Positive rate (%)	
Task	Train	Valid.	Test	Query	MIMIC	EHRSHOT
MT	4,564	1,521	1,523	4,403	4.79	6.56
LLOS	4,564	1,521	1,523	4,403	32.24	27.00
RA	4,565	1,521	1,523	4,404	2.30	9.60
PHENO	4,267	1,421	1,411	4,098	12.58	19.08
UTI	12,905	4,296	4,304	3,753	11.55	12.76
SEPSIS	13,016	4,368	4,391	3,991	8.20	18.04
PNA	12,900	4,310	4,333	3,907	12.25	18.99
MI	13,257	4,406	4,403	3,940	3.59	10.00
FX	12,749	4,254	4,254	3,807	5.67	7.20

Counts and rates are computed from the materialized inputs. Each task has one instance per patient, but patients can appear in multiple task rows. Positive rates describe MIMIC-IV test and EHRSHOT query instances; for PHENO, they count positive instance–label cells.

C.3.1 MI LABEL CONSTRUCTION

The MI task in the main nine-task comparison uses the revised label construction described here. This construction was called MI-ACUTE in the earlier audit; we report it as MI in the main comparison.

The original defining set contained 201 concepts, of which eight denoted prior infarction (*old myocardial infarction* and its anatomical variants, and a code reading "[old] or [healed] or [personal history of]"). These concepts can identify a history of infarction rather than the event targeted by the task. In the earlier audit, they also placed a deterministic positive marker in the input of 35.5% of cases under C2 (Appendix F.6), the highest rate of any task. We therefore exclude these eight concepts from the label-defining set and re-derive the cohort.

In that audit, this revision reduced the case count from 2,818 to 1,458 and the C2 flag rate from 35.5% to 12.8%, before the separate C2 correction was applied. These figures describe the audit cohort, not the current evaluation cohort, whose counts are reported in Table 7. The MI-ACUTE rows in the audit tables in Appendix F retain their historical name and measurements; applying this label revision does not by itself establish that the separate C2 input correction was enabled in the current build.

C.3.2 PHENO LABELS FROM SOURCE CODES

Following Harutyunyan et al. (2019), PHENO is defined as the set of single-level CCS categories present in an admission, restricted to the 24 categories of the released phenotype list and identified

by CCS category number. The labels are determined directly from the original diagnosis codes of the condition records.

The 24 categories are not equally common, and the two institutions do not agree on which are common. Table 8 gives the per-label positive rates. Five categories differ by at least a factor of two, led by *other lower respiratory disease* (7.09% at the source against 41.65% at the target) and *pleurisy; pneumothorax; pulmonary collapse* (6.87% against 28.65%). The target is at least as common in 21 of the 24 categories; the exceptions are *acute myocardial infarction*, *coronary atherosclerosis* and, marginally, *chronic obstructive pulmonary disease*, and none of the three falls below 0.80 of the source rate. A single aggregate positive rate therefore hides most of the institutional difference this task carries.

Table 8: Per-label positive rates for PHENO, over MIMIC-IV test instances ($n = 1{,}411$) and EHRSHOT query instances ($n = 4{,}098$). "ratio" is target over source; rows where the two institutions differ by at least a factor of two are set in bold. The cell-level averages of these columns, 12.58 and 19.08, are the PHENO entries of Table 7.

CCS category	MIMIC-IV (%)	EHRSHOT (%)	ratio
Acute and unspecified renal failure	14.81	16.94	1.14
Acute cerebrovascular disease	3.40	5.30	1.56
Acute myocardial infarction	8.08	6.44	0.80
Cardiac dysrhythmias	23.67	37.48	1.58
Chronic kidney disease	13.39	19.35	1.44
Chronic obstructive pulmonary disease	9.64	9.57	0.99
Complications of surgical/medical care	13.61	17.30	1.27
Conduction disorders	**3.76**	**14.67**	3.90
Congestive heart failure; nonhypertensive	17.22	17.86	1.04
Coronary atherosclerosis and related	25.02	22.25	0.89
Diabetes mellitus with complications	10.77	12.01	1.11
Diabetes mellitus without complication	12.47	21.82	1.75
Disorders of lipid metabolism	31.54	35.11	1.11
Essential hypertension	27.36	37.07	1.35
Fluid and electrolyte disorders	19.70	28.33	1.44
Gastrointestinal hemorrhage	2.76	4.61	1.67
Hypertension with complications	16.09	21.21	1.32
Other liver diseases	**7.30**	**15.08**	2.07
Other lower respiratory disease	**7.09**	**41.65**	5.88
Other upper respiratory disease	**2.98**	**7.52**	2.52
Pleurisy; pneumothorax; pulmonary collapse	**6.87**	**28.65**	4.17
Pneumonia	5.88	10.44	1.78
Respiratory failure; insufficiency; arrest	10.63	15.54	1.46
Septicemia (except in labor)	7.94	11.64	1.47

Both MIMIC-IV and EHRSHOT contain ICD-9-CM and ICD-10-CM diagnoses, so we use the official crosswalk of each edition (AHRQ's Single-Level CCS for ICD-9-CM and CCS for ICD-10-CM) (Healthcare Cost and Utilization Project (HCUP)). HCUP releases the two editions with matched category numbers, so the outputs of the two crosswalks live in the same category space.

A record's code is normalized by trimming, upper-casing and deleting the decimal point, and its code set is taken from the vocabulary of the record's source concept (`ICD9CM` or `ICD10CM`). The join is an exact match on the pair (code, code set), so a code is compared only against the crosswalk of its own edition. The label vector of an instance is the set of categories carried by the diagnoses attributed to its target admission. Figure 3 traces one category from records to label.

code	code set	normalized	crosswalk	CCS
'410.71'	ICD-9-CM	41071	ICD-9-CM	100
'410.11'	ICD-9-CM	41011	ICD-9-CM	100
'410.72'	ICD-9-CM	41072	ICD-9-CM	100
'I21.4'	ICD-10-CM	I214	ICD-10-CM	100
'I21.A1'	ICD-10-CM	I21A1	ICD-10-CM	100
'I21.09'	ICD-10-CM	I2109	ICD-10-CM	100

Figure 3: Label construction for CCS category 100 (*acute myocardial infarction*). Each diagnosis code attributed to the target admission is normalized, matched against the crosswalk of its own code set, and contributes its category to the label vector; six codes from MIMIC-IV are shown.

C.4 Record eligibility and preprocessing scope

Each prediction instance has its own prediction time. Only records with a recorded timestamp strictly before that time are eligible as event inputs. This cutoff is distinct from the clinical availability of a retrospectively recorded diagnosis or billing code. The existing timestamp and exclusion audits are in Appendix F; their proposed corrections must not be assumed to have been enabled in the current data build.

TabEHR fits concept-wise numerical normalization on source training data and reuses the fitted statistics at the target institution. Record construction and event encoding follow section 3.3; temporal binning and coverage-first selection are separate operations, with their measurements and build-reproducibility checks in Appendix E. These processing choices are not imposed on every comparator: the main MEDS-Tab-style baselines use raw-value statistics, and sequence models retain their own input processing (Appendix D). Source-fitted preprocessing is also distinct from support-dependent centering of the learned anchor readouts.

C.5 Evaluation data access and support sets

Internal prediction. Models are trained or fitted on the MIMIC-IV downstream training partition and selected on its validation partition. Queries are held-out MIMIC-IV test instances. In-context predictors use the complete eligible training set of the same task as support, rather than the smaller episodes used during aggregator training. Query labels are used for evaluation only.

Cross-institution prediction without target labels. The selected source models predict the corresponding nine EHRSHOT tasks without parameter updates. In-context predictors retain the same MIMIC-IV support used for internal prediction. TabEHR's support-centering mean is computed from that source support. EHRSHOT outcome labels are used only to calculate metrics, not for model selection, prediction context, or preprocessing fits. Target query records are read as inference inputs, without fitting patient-level normalization statistics to those queries.

Target concept identifiers and vocabulary metadata are available. The cached embedding table and the covariance used for whitening include target vocabulary entries; the construction of these artifacts is described in Appendix B.1. This metadata access does not change the source-only candidate pool for anchor construction (Appendix C.2). We therefore describe the setting as *target-label-free transfer with target vocabulary access*. The coverage audit and whitening access policy are described in Appendix G. Coverage was measured on the separate fifteen-task benchmark inputs, not the nine-task cohorts above.

C.6 Nine-task fitting and model selection

Patient-prediction models are fitted on the MIMIC-IV downstream training partition and selected using source validation data only. Sequence models select checkpoints separately for each task by validation AUROC; XGBoost hyperparameters are also selected by task-specific validation AUROC. TabEHR and its concept-embedding variants select one checkpoint per seed by the highest macro AUROC across the nine validation tasks, and use that checkpoint for all tasks at both institutions. The statistical TFM has no task-specific checkpoint to train; its input-feature choices use source validation data. EHRSHOT outcome labels are excluded from all these selection decisions.

The comparison uses the common seed set $\{100, 200, 300\}$ and reports means of seed-specific metrics. Selected checkpoints, available-run exceptions, and the scope of stochastic variation are documented in Appendix B.3; method-specific fitting procedures are provided in Appendix D.

C.7 NINE-TASK METRICS AND UNCERTAINTY

We report AUROC and area under the precision–recall curve (AUPRC) for each task. For PHENO, we first calculate each binary label's metric and macro-average the eligible labels; a label must contain both classes in the support/training and evaluated query sets. The overall result is the unweighted mean across the nine tasks, with PHENO contributing one task-level value. Thus, tasks with more patients or more phenotype labels do not receive more weight in the reported macro-average.

Positive rates are reported separately for each evaluation cohort (Table 7). We compare methods within the same task and cohort; an AUPRC difference between institutions also reflects their different outcome prevalences. The available provenance does not yet establish identical implementation details for tied scores and phenotype-label eligibility across every method-specific evaluator.

For repeated training runs, we average seed-specific metrics rather than first averaging probabilities into an ensemble. Between-seed variation for TabEHR is reported in Appendix J. Patient-level paired bootstrap intervals have not been computed; seed variability should not be interpreted as evaluation-sample uncertainty.

D BASELINE IMPLEMENTATION DETAILS

D.1 SEQUENCE BASELINES

This appendix records the sequence-baseline reimplementations under the common data construction and evaluation protocol of Appendix C. The reported runs use batch size 32 with an optimizer step every four batches, learning rate 5×10^{-5}, at most 50 epochs, and early stopping with patience 5 on validation loss. For the current comparison, checkpoints are selected separately for each task and seed by the highest MIMIC-IV validation AUROC.

Sequence implementations. We compare BEHRT (Li et al., 2020) and CDM-BERT (Kim et al., 2025a), as well as BEHRT equipped with MedTok (Su et al., 2025) or MedRep (Kim et al., 2025b) concept representations. The sequence baselines undergo EHR pretraining on the reserved source pretraining patients, followed by task-specific supervised finetuning on the downstream training split. The split and training details follow below.

TabEHR embs. + BEHRT uses TabEHR concept embeddings within the BEHRT sequence pipeline. These concept-description embeddings come from a BiomedBERT encoder trained with InfoNCE, without the ontology-graph training or knowledge distillation used by MedRep. It follows the same v4 patient pretraining partition and finetuning protocol as MedRep + BEHRT, replacing only the concept vectors while retaining BEHRT's event-input construction. This row tests concept representation within the sequence pipeline.

BiomedBERT comparator. BiomedBERT + BEHRT uses a public pretrained BiomedBERT checkpoint (Gu et al., 2020) (abstract + full-text variant, revision `e1354b7a`) to encode OMOP concept names once, without further training, and supplies the last-layer [CLS] vectors to the BEHRT representation interface used by the MedRep and MedTok variants. Value-bin tokens are encoded as the concept name followed by their decile index, special-token rows are shared with the MedRep variant, and every vocabulary entry had a concept name. As in MedRep + BEHRT, the concept embeddings are frozen during EHR pretraining and all embedding layers are frozen during task finetuning. EHR pretraining and task finetuning use the same source partitions, training budgets, and source-validation selection rule as the corresponding BEHRT controls; the BEHRT encoder is pretrained anew with these vectors rather than reusing another variant's checkpoint. This is an adaptation of a released text encoder to our EHR pipeline; it does not reproduce biomedical language-model pretraining from scratch.

Baseline splits. All sequence baselines use the source partitions in Appendix C.2, including the reserved 258,632-patient pretraining partition and the frozen downstream assignment. Released split files are replaced with these common identifiers, so pretraining and downstream patients remain disjoint.

D.2 STATISTICAL-FEATURE BASELINES

Feature construction and selection. We implement a MEDS-Tab-style feature table (Oufattole et al., 2024) with nested lookback windows of 1, 7, 30, and 365 days and the complete available history. For each concept and window, the main comparison uses six aggregations: record count, numerical-value count, sum, sum of squares, minimum, and maximum. It uses raw recorded values, without TabEHR's unit, range, or z-normalization pipeline and without ontology expansion. Mean, last value, and days since the last record belong to the alternative statistical pipeline in Table 32. The main statistical pipeline is fixed independently of external outcomes.

Features are pruned at 1% source-training coverage and ranked by source-fitted XGBoost gain, with source validation used for early stopping. We retain $K = 512$ statistical columns plus age and sex, and keep their identities and order fixed at the target. XGBoost (Chen & Guestrin, 2016) fits trees on this feature table. The second predictor sends the identical table through the scalar encoder of the frozen tabular foundation model, with its event input path disabled, and uses the source training rows as support. Feature construction and column selection still use source data even though the foundation model itself receives no parameter updates.

Predictor fitting and selection. For XGBoost, maximum depth $\{3, 5, 7\}$ and minimum child weight $\{1, 5\}$ form a six-setting grid selected on source-validation AUROC. The statistical TFM requires no task-specific parameter fitting. Feature selection and predictor selection exclude EHRSHOT outcome labels.

Numerical precision. The saved main-run configuration enables BF16 mixed-precision training and sets `use_amp_eval=true`. A single evaluation precision has not been verified across all in-context runs; the saved training configuration alone does not establish it.

D.3 Concept-replacement controls with the tabular predictor

MedRep + TFM and BiomedBERT + TFM replace the concept embeddings while retaining TabEHR's record construction, event channels, anchor-aggregation architecture, and frozen tabular backbone. The patient-level aggregator is retrained for each representation. These names identify the concept source within the shared event-to-cell pipeline, whose aggregated vector cells enter the predictor. This comparison tests the choice of concept representation under the same aggregation and prediction architecture. These comparisons do not separately isolate encoder quality from any differences in concept coverage.

MedRep + TFM construction. MedRep concept vectors (Kim et al., 2025b) replace the embedding table under the same procedure, including the common-scale step below; the required factor is 0.381, and the anchor set and its temperature are rebuilt from the rescaled table. Its aggregators had not finished training when this version was prepared, and the published snapshot uses step $4{,}000$ for all three seeds. The latest source-validation selection for seed 100 moved to step $5{,}000$, but its corresponding internal and external evaluations were not yet present in the inspected artifacts; the published row is therefore retained without mixing selection and evaluation snapshots. The row is an interim result and must be re-evaluated after training; further training does not guarantee an increase in external performance.

BiomedBERT + TFM construction. Concept names are encoded with the public BiomedBERT checkpoint (Gu et al., 2020) and the final-layer `[CLS]` vector is taken as the concept embedding, over the same $39{,}259$ concept identifiers and the same row order as the TabEHR concept embedding table. The event arrays of the record store are shared by reference with the main pipeline, so the concept embedding table is the only differing input; a byte-level check over the evaluation sets found no other difference. Because the whitening ridge is absolute, every substituted table is placed on a common scale: its median covariance eigenvalue is matched to that of the TabEHR concept embedding table, so the ridge carries the same relative weight (2.1%) in both arms. For BiomedBERT this scale already agreed and the required factor was unity, so its embeddings are used unchanged. The whitening transform and the anchor set are rebuilt from that table, and the anchor temperature is refit to the same target assignment entropy as the main TabEHR configuration. The aggregator is then retrained from a random initialization under the same training budget as the main run; initializations are drawn independently per run in both arms, so the seed-to-seed spread reported for TabEHR applies here as well. Checkpoint selection follows the rule in Appendix C.6.

E RECORD CONSTRUCTION AND TOKEN SELECTION

This appendix reports the measurements behind the bucketing choice in section 3.3. Unless a sample size is stated, counts are exhaustive over the named cohort. Samples are drawn without replacement from the instance table of the stated task and institution.

E.1 THE GRID

Bin edges are generated by equation 7 with $\delta = 0.5$, rounded to whole days and deduplicated, then prepended with 0:

$$0,\ 1,\ 2,\ 3,\ 5,\ 8,\ 11,\ 17,\ 26,\ 38,\ 58,\ 86,\ 130,\ 195,\ 292,\ 438,\ 657,\ 985,\ 1478,\ 2217,\ 3325$$

This is a prefix of the grid, not a lookback restriction: edges continue for older records. The scale $T_0 = 3651$ days in the recency encoding does not clip history. The first three intervals are one day wide; subsequent intervals widen geometrically after rounding and deduplication.

A token stores the contributing record count, elapsed time of the most recent record, most recent non-missing normalized value, and within-bin minimum and maximum. Optional consecutive-bin differences are stored separately and are not active inputs in the main configuration. In the audited store version, an instance was capped at $8{,}192$ tokens before applying the model budget. That cap did not bind on the audited MT tables: maxima were $4{,}541$ at the source and $3{,}763$ at the target. These version-specific measurements describe the audited store; the model retains at most $L = 512$ tokens by coverage-first selection.

E.2 BUCKETING AGAINST ALTERNATIVE GRANULARITIES

Table 9 compares three units on identical instances. Table 10 gives the distribution the compression acts on: the share of tokens falling in each bin, and the number of raw records each token absorbs.

Table 9: Median tokens per instance under three granularities, and the fraction of instances above the budget $L = 512$. Each cell samples $1{,}500$ prediction instances drawn from the task's full instance list, without restriction to an evaluation split, and counts only records whose concept appears in the shared vocabulary.

		median tokens per instance			instances above L	
Institution	Task	records	(concept, day)	(concept, bin)	(concept, day)	(concept, bin)
MIMIC-IV	MT	197	160	146	22.5%	**10.7%**
MIMIC-IV	PHENO	176	150	138	23.9%	**11.6%**
EHRSHOT	MT	2,848	438	285	46.9%	**31.1%**
EHRSHOT	PHENO	3,178	506	309	49.5%	**33.5%**

Two readings matter. First, compression rises monotonically with bin width away from the prediction time — from 1.26 to 4.76 records per token at the source over the first 58 days — which is the intended behaviour: distant history is summarized more coarsely than recent history without any hand-set schedule. Second, the same grid compresses the target far harder than the source in the most recent bin (19.87 against 1.44). The two institutions differ by an order of magnitude in records per patient-day, and a fixed grid expressed in elapsed time rather than in records absorbs that difference. Under a record-level encoding the target's median instance carries $2{,}848$ tokens against the source's 197; under the bin grid, 285 against 146.

E.3 HOW LONG THE STORED RECORDS ARE, BEFORE AND AFTER SELECTION

The tables above sample instances, so we also state the exhaustive counts. Every instance in the nine task stores is measured at two stages: the number of (concept, bin) tokens the build materializes, and the number that survive the budget $L = 512$. Table 11 reports the first stage together with the fraction that the budget binds on; the second stage is the first capped at 512, so its median equals the first wherever the median lies below the budget, which it does for every task at both institutions.

Table 10: Token occupancy of the ten most recent bins. Both columns pool the sampled instances rather than averaging per-instance ratios: "share" is the percentage of all sampled tokens that fall in the bin, and "raw/token" is the total number of records absorbed by those tokens divided by their count. Samples of 3,000 instances drawn from the task's full instance list; PHENO uses the restricted-admission cohort.

	MIMIC-IV				EHRSHOT			
	MT		PHENO		MT		PHENO	
Bin start (d)	share	raw/tok	share	raw/tok	share	raw/tok	share	raw/tok
0	**23.4%**	1.44	**20.3%**	1.56	**14.0%**	**19.87**	**11.6%**	**23.84**
1	1.0%	1.26	0.9%	1.19	3.4%	1.30	2.4%	1.33
2	0.8%	1.59	0.8%	1.38	1.0%	1.45	1.1%	1.69
3	1.1%	2.09	1.1%	1.84	1.5%	3.07	1.7%	3.16
5	1.9%	2.50	1.8%	2.03	2.2%	3.73	2.3%	3.85
8	1.8%	2.77	1.7%	2.70	2.1%	4.57	2.1%	6.16
11	2.9%	3.42	2.9%	3.21	3.0%	5.38	3.2%	6.10
17	3.4%	3.79	3.5%	3.97	3.6%	7.25	3.8%	6.67
26	3.5%	4.34	3.8%	4.32	3.7%	7.54	3.8%	7.61
38	4.6%	4.76	4.8%	4.88	4.6%	9.45	4.8%	9.34

Table 11: Stored tokens per instance, over every instance in each task store rather than a sample. "median" and "p90" are over instances; "$> L$" is the fraction whose stored length exceeds the budget $L = 512$. MIMIC-IV columns cover all partitions; the source and target counts of evaluated instances are in Table 7.

	MIMIC-IV				EHRSHOT			
Task	instances	median	p90	$> L$	instances	median	p90	$> L$
MT	7,608	140	562	11.4%	4,403	292	1,147	32.8%
LLOS	7,608	140	562	11.4%	4,403	292	1,147	32.8%
RA	7,609	257	749	21.0%	4,404	410	1,095	39.1%
PHENO	7,099	131	558	11.2%	4,098	293	1,151	32.7%
UTI	21,505	162	573	12.4%	3,753	287	945	27.4%
SEPSIS	21,775	162	570	12.2%	3,991	295	1,000	29.2%
PNA	21,543	156	536	11.0%	3,907	289	1,003	29.0%
MI	22,066	165	596	13.2%	3,940	300	983	30.2%
FX	21,257	173	613	13.8%	3,807	290	963	29.1%

Target instances carry about twice the stored tokens of source instances at the median and reach the budget two to three times as often. The gap is a property of what each institution records over the eligible window, and it is present before any selection rule applies.

E.4 FILLING THE BUDGET: COVERAGE-FIRST VERSUS RECENCY

Coverage-first is defined in equation 8; this subsection reports the measurements that motivated adopting it over recency. The budget $L = 512$ binds for a minority of instances at the source and roughly a third at the target: 11.4% (MT), 21.0% (RA), 13.2% (MI) and 11.2% (PHENO) of source instances exceed it, against 32.8%, 39.1%, 30.2% and 32.7% of target instances. Table 12 compares the two fill rules on exactly those instances; rows within budget are unaffected by the choice.

In Table 12, H is the oldest retained token's age in days; coverage is the fraction of concepts retaining a token; tokens/concept is the mean multiplicity among retained concepts. The final two columns give the budget share assigned to tokens older than one year and to the most recent bin. These over-budget instances are longer than typical histories; cohort medians are reported in Table 9.

The trade is between depth and breadth, and it is sharper at the target. Recency buys trajectory: more tokens per concept, and a higher share of tokens whose value channel is populated (52.7% against 46.3% at the source, 45.7% against 33.8% at the target), because recent measurements are more

Table 12: Coverage-first versus recency selection for records exceeding $L = 512$ tokens. Entries are medians over up to $1{,}500$ sampled instances per cell.

Institution	Task	Rule	H (d)	coverage	tokens/concept	$\geq$ 1y	final 24h
MIMIC-IV	MT	recency	306	0.73	2.37	0.0%	13.3%
MIMIC-IV	MT	coverage-first	1,120	**1.00**	1.61	20.5%	13.3%
MIMIC-IV	PHENO	recency	305	0.73	2.40	0.0%	10.7%
MIMIC-IV	PHENO	coverage-first	1,104	**1.00**	1.63	20.4%	10.7%
MIMIC-IV	RA	recency	48	0.74	2.43	0.0%	5.1%
MIMIC-IV	RA	coverage-first	349	**1.00**	1.72	0.0%	5.1%
MIMIC-IV	MI	recency	121	0.74	2.56	0.0%	10.9%
MIMIC-IV	MI	coverage-first	714	**1.00**	1.82	6.3%	10.9%
EHRSHOT	MT	recency	247	0.64	2.57	0.0%	15.6%
EHRSHOT	MT	coverage-first	1,929	**1.00**	1.62	22.3%	15.6%
EHRSHOT	PHENO	recency	265	0.63	2.64	0.0%	12.5%
EHRSHOT	PHENO	coverage-first	1,952	**1.00**	1.63	23.0%	12.5%
EHRSHOT	RA	recency	70	0.69	3.03	0.0%	8.2%
EHRSHOT	RA	coverage-first	1,273	**1.00**	2.00	5.3%	8.2%
EHRSHOT	MI	recency	144	0.67	2.86	0.0%	8.0%
EHRSHOT	MI	coverage-first	1,560	**1.00**	1.87	10.0%	8.0%

likely to carry a comparable value. Coverage-first buys history: every concept is represented at least once, the retained horizon lengthens by a factor of 3.7 at the source and 7.8 at the target, and the composition shifts away from measurements towards conditions, drugs and procedures (measurement share $0.72 \to 0.64$ at the source, $0.66 \to 0.53$ at the target). The number of concepts retaining a usable trajectory — at least two tokens — falls from 51 to 43 at the source.

The share of the budget spent on the most recent bin is identical under both rules to the displayed precision in Table 12. Coverage-first therefore changes the breadth and depth of retained history without resolving the timestamp concentration described above. A concept omitted by selection cannot contribute its own event content to any anchor. Related retained concepts can still influence the same anchors, so dropping a concept does not force a corresponding slot to zero or to its support mean. We choose broader concept coverage under the fixed budget; predictive consequences require the controlled comparison rather than following from this coverage statistic alone.

E.5 VALUE-PROCESSING REPRODUCIBILITY

A reconstruction audit identified nondeterminism in the original materialized store: values from same-time duplicate records could be aggregated while a unit label was selected independently. Mixed-unit records can therefore change normalized values without changing event identities, counts, or timestamps. Selecting a unit label deterministically alone does not resolve the mismatch between values and units.

A revised builder was used in exploratory binning experiments, but those stores are not interchangeable with the original main-table inputs. Accordingly, we do not report that experiment as a matched ablation of the main model. The reported main results refer to the original materialized inputs; this audit does not establish that rebuilding them reproduces every numerical value.

E.6 AUDIT NOTE ON TARGET-SIDE MEASUREMENTS

The target-side columns of this appendix are reported as an audit of the design, not as inputs to it. The grid of equation 7, the cap of $8{,}192$, the budget $L = 512$ and the coverage-first rule are fixed by δ and two integers, were set on source data before the target extract was tokenized, and are not selected per institution, per task or per cohort. No quantity in Table 10 or Table 12 enters Ω (section 4.1).

F Input Availability and Construction Audits

Current dataset definitions, cohort sizes, and evaluation rules are collected in Appendix C. This appendix preserves construction audits, including measurements from earlier eight-task and pretext-pool lineages. Unless a measurement is explicitly matched to the current inputs, it is not a replacement for the nine-task counts in Appendix C.2 and Appendix C.3. The status of proposed corrections must be checked against the final build manifest.

F.1 Concept-to-domain functionality

Each event inherits the domain of the OMOP table from which its record was read, rather than a domain looked up from the concept vocabulary. Whether the domain is a function of the concept is therefore a property of the data, not of the vocabulary, and was measured on the historical pools listed below (Table 13). A concept is counted as violating the property if it occurs under more than one domain anywhere in the pool, within a task or across tasks.

Table 13: Concepts occurring under more than one OMOP domain in the historical audited pools. No listed pool contains such a concept.

Pool	Distinct concepts	Events	Multi-domain concepts
Self-supervised pretraining (85% cohort)	23,843	46,938,343	0
Pretext episodes (85% cohort)	26,411	121,504,658	0
Downstream training (15% cohort, 8 tasks)	16,387	16,796,117	0

F.2 Temporal boundary

The current recorded-time cutoff is specified in Appendix C.4. The build accounting of the audited stores records that strict inequality; this verifies the timestamp boundary, not the clinical availability of every record type.

F.3 Which record types carry an intrinsic time

Records without an intrinsic event time are inserted in batches and therefore share timestamps within an encounter, whereas records generated by the act they describe do not. Table 14 measures this directly: within each encounter, the number of distinct timestamps divided by the number of records of that type. A value near 1 indicates per-record timing; a low value indicates batch insertion.

Table 14: Timestamp granularity by OMOP record type, source institution. Ratio is the per-encounter mean of (distinct timestamps)/(records), over encounters with at least three records of that type. Samples of 1.5M records per table.

Table	Type	Records	Distinct-time ratio
condition	EHR	812,890	0.994
condition	EHR billing record	687,110	**0.285**
procedure	EHR	978,558	0.809
procedure	EHR order	420,619	0.567
procedure	EHR billing record	100,823	**0.309**

The separation is clean: billing-typed records are batched at both tables, and EHR-typed conditions are not. This is the operational test we use to decide which types require an availability assignment, independently of the type label itself.

F.4 C1: composition of the MT/LLOS input window

Table 15 decomposes the input window of the admission-anchored tasks by record type and by whether the record is attributed to the observed admission. Attribution uses the OMOP encounter

identifier, not a time heuristic: we identify the admission whose start date equals $T - 1$ day and compare encounter identifiers. The cohort's median length of stay is 4 days (mean 7.1), so for most instances the observed admission continues well past T.

Table 15: Records inside the MT/LLOS input window (7,612 instances). "Current" is attribution to the observed admission by encounter identifier. The drug and measurement tables carry intrinsic times and are listed for composition only.

Table	Type	In window	Current encounter	Share
condition	EHR billing record	213,487	**89,625**	**42.0%**
condition	EHR	160,971	12,946	8.0%
procedure	EHR	177,670	11,250	6.3%
procedure	EHR order	107,744	35,281	32.7%
procedure	EHR billing record	22,395	**0**	**0.0%**
drug	(all)	565,351	—	—
measurement	(all)	9,220,664	—	—

Three observations follow.

First, the leakage is concentrated in exactly the type that Table 14 identifies as batched: 89,625 discharge-coded diagnoses of the ongoing admission sit inside the window, and 7,586 of 7,595 patients (99.9%) carry at least one.

Second, the majority of billing-typed diagnoses in the window — 123,862 of 213,487 (58.0%) — come from previously completed encounters and are legitimate history. A rule that removed all billing-typed records would discard more than it should. This is why attribution, not type alone, defines the correction.

Third, billed procedures show the opposite pattern: all 22,395 in the window belong to prior encounters, so this table contributes no C1 leakage at these tasks. DRG codes and discharge disposition never enter the input, because the event cache is built from four OMOP tables (condition, drug, measurement, procedure) only.

A single instance. Table 16 shows one row for which the mechanism is unambiguous. Two of the batched codes — *ventilator-associated pneumonia*, which by definition requires at least 48 hours of mechanical ventilation, and *acute renal failure due to tubular necrosis* — cannot have been established on the day they are stamped. The batch is the discharge diagnosis list of a 39-day admission written back to the admission timestamp.

Table 16: One MT/LLOS instance. Admission 2154-01-19 to 2154-02-27 (39 days, died); $T =$ 2154-01-20 00:00; MT$= 1$, LLOS$= 1$. All 49 condition records in the input window fall on the admission day; this patient has no earlier condition history, so the entire condition input is the discharge summary of the stay being predicted.

Timestamp	Type	Count
2154-01-19 13:27:00 (single)	EHR billing record	38
11 distinct times, 13:00–23:5x	EHR	11
Selected batched codes: sepsis; shock; ventilator-associated pneumonia; acute renal failure due to tubular necrosis; toxic encephalopathy; acute hypoxemic respiratory failure; acquired coagulation factor deficiency.		

F.5 C1: THE CORRECTION AND ITS PER-TASK FORM

We adopt an availability-aware censoring rule.

> *Availability-aware censoring.* An event enters $\mathcal{E}_i$ only if it was clinically or administratively available before T_i. For retrospective encounter-level records without

an intrinsic event timestamp — discharge diagnoses, billed procedures and DRG codes — availability is the discharge time of the encounter they belong to, not its admission time.

Applied per task (Table 17), this keeps prior history everywhere and removes only what the prediction time cannot yet see.

Table 17: Allowed history per task under the availability-aware censoring rule.

Task	T	Discharge-coded records of the current encounter
MT, LLOS	admission + 1 day	excluded
RA	discharge + 1 day	allowed (already finalized)
FX, PNA, SEPSIS, UTI, MI	index event − 1 day	excluded if the encounter is still open at T

RA functions as an internal control. Its input contains the same discharge-coded records, but its prediction time follows discharge, so those records are genuinely available; correspondingly only 0.27% of its window's billing-typed diagnoses fall in the final day. The defect is therefore one of timestamp semantics, not of the records themselves.

Relation to the implemented correction. The build used in this paper removes billing-typed conditions falling in the final 24 hours before T, which for MT/LLOS is the admission day. Table 18 compares that heuristic against encounter attribution. It captures every current-encounter record ($89{,}625$ of $89{,}625$) and additionally removes 908 conditions and 582 procedures that belong to prior encounters and should have been kept — 0.4% of the removed conditions. We report the heuristic because it is what the released build used, and note the exact residual here.

Table 18: Implemented 24-hour heuristic versus encounter attribution, MT/LLOS.

Table	Attribution rule	24-hour rule	Both	Heuristic only (over-removed)
condition (billing)	89,625	90,533	89,625	908
procedure (billing)	0	582	0	582

Open item. Procedure orders attributed to the current encounter ($35{,}281$, 32.7% of that type) remain in the input. An order carries an intrinsic order time, so it is plausibly available at T; its distinct-time ratio of 0.567 (Table 14) is nevertheless below the EHR-typed records and warrants a separate check. We flag this rather than assert leakage.

F.6 C2: THE DEFINING CONCEPT INSIDE THE INPUT WINDOW

Let $\mathcal{C}_t$ be the set of concepts used to define the positive label for disease task t. Table 19 reports, for each disease task, the fraction of cases and controls whose input window contains a record from $\mathcal{C}_t$. The control figure is 0.00% by construction, since a control never holds such a record. Membership in $\mathcal{C}_t$ inside the window is therefore a deterministic positive predictor. We quantify the floor as the AUPRC of a ranker placing all flagged instances first and ordering the remainder at random.

One instance illustrates the mechanism: a SEPSIS case with T = 2140-01-19 carries 22 records from $\mathcal{C}_{\text{SEPSIS}}$ inside its window, spread over thirteen years (2127–2137), including the codes *sepsis* and *severe sepsis* verbatim. Note that C2 is independent of C1: it follows from the case/control asymmetry in the label definition, not from timestamp semantics.

Correction. In the historical corrected builds, records whose concept lies in $\mathcal{C}_t$ were removed from the input of task t at both institutions: 681–$3{,}001$ records per task at the source and $2{,}957$–$8{,}407$ at the target. A post-build check confirmed zero remaining occurrences in those builds. The same check remains to be linked to the current nine-task input manifest.

Table 19: Label information inside the input window, source institution.

Task	Cases flagged	Controls flagged	Prevalence	Flag-only AUPRC	Ratio
MI	35.5%	0.00%	—	—	—
UTI	23.1%	0.00%	0.138	0.315	2.29×
PNA	20.0%	0.00%	0.144	0.295	2.05×
SEPSIS	15.9%	0.00%	0.100	0.231	2.31×
MI-ACUTE	12.8%	0.00%	0.056	0.171	3.07×
FX	11.2%	0.02%	0.111	0.199	1.80×

G VOCABULARY COVERAGE AND ACCESS

This appendix separates measured vocabulary coverage from the metadata access used to construct the main model's semantic geometry.

G.1 COVERAGE UNDER VOCABULARY ADMISSION GATES

The vocabulary access specified in Appendix C.5 permits target concept identifiers and metadata. The measurements below use the separate fifteen-task EHRSHOT benchmark inputs. They describe vocabulary admission, not few-shot predictive performance or coverage measured on the nine-task cohorts. No target labels are needed to count represented input records.

Two gates can decide which target records become event tokens. The *source-only* gate admits a record if its concept is one the source institution records — 30,948 concepts with a representation. The *transductive* gate, which is the one we use, admits a record if its concept appears in the target's own vocabulary and has a representation — 21,684 concepts. The two sets overlap in only 13,373 concepts: 8,311 of the target's concepts are outside the source gate, and 17,575 source concepts never appear at the target. The union, 39,259 concepts, is the space the anchors are whitened in.

Table 20 reports, over all fifteen benchmark tasks, the share of pre-prediction records each gate represents. The transductive gate represents 94% of records and the source-only gate 65–70%; the difference is concentrated in the two domains that carry the most records. Under the source-only gate only 13–17% of distinct measurement concepts survive, which is the mechanism: a laboratory analyte recorded at the target under a concept the source never used becomes invisible, and measurements are where the value channel lives. The overall record-coverage fraction varies by at most one percentage point from the group mean under either gate; this does not establish equal effects on prediction across tasks.

Table 20: Share of pre-prediction target records represented, by domain and gate, over the fifteen benchmark tasks. Ranges are across tasks. "Concepts" counts distinct concepts rather than records. The transductive gate is the one used in this coverage audit.

	Gate	Condition	Drug	Measurement	Procedure	All
Records	transductive (21,684)	.89–.93	1.00	.95–.96	.06–.09	**.94**
	source-only (30,948)	.84–.85	.68–.71	.65–.70	.05–.07	**.65–.70**
Concepts	transductive	.95–.96	1.00	.95–.96	.43–.55	—
	source-only	.81–.85	.28–.36	.13–.17	.43–.48	—

Procedure coverage is low under both gates: only 6–9% of records are represented, compared with 43–55% of distinct procedure concepts. Thus, frequently recorded procedure concepts account for much of the lost coverage. Procedures comprise 1.7% of this audited record stream, so high overall coverage does not establish adequate procedural-history coverage.

G.2 WHITENING AND TARGET VOCABULARY ACCESS

The main model estimates the partial-whitening transform from the cached union of source and target concept embeddings (39,259 concepts). Anchor candidates and their construction frequencies come from the source pretraining pool (Appendix C.2). Thus, target vocabulary metadata affect the coordinate transformation, while target outcome labels and target patient-level statistics are not used to fit it.

No completed matched comparison in the current configuration establishes that source-only whitening gives equivalent predictive performance. The reported transfer results therefore retain the vocabulary-access condition specified in Appendix C.5. Clinical retrieval protocols and scored-query coverage are reported in Appendix H.2.

Table 21: Pairwise relation discrimination using cosine similarity. n is the number of positive relation pairs; the same number of domain-matched negative pairs is used.

Relation (target domain)	n	BiomedBERT	MedRep	TabEHR concept space
RxNorm has ing (Drug)	2,000	0.654	0.960	**0.995**
Contained in panel (Measurement)	2,000	0.738	0.867	**0.960**
Has due to (Condition)	2,000	0.696	0.765	**0.921**
Consists of (Drug)	2,000	0.953	**0.996**	0.992
Has focus (Condition)	2,000	0.792	0.755	**0.975**
Followed by (Procedure)	2,000	0.680	0.772	**0.956**
Occurs after (Procedure)	1,424	0.637	0.825	**0.960**
Mean		0.736	0.848	**0.966**

Table 22: Mean cosine similarity for positive and domain-matched negative relation pairs. Values in parentheses show the positive–negative separation.

Relation	BiomedBERT	MedRep	TabEHR concept space
RxNorm has ing	0.958 / 0.949 (0.008)	0.949 / 0.815 (0.134)	0.747 / 0.208 (**0.539**)
Contained in panel	0.968 / 0.954 (0.013)	0.933 / 0.828 (0.105)	0.527 / 0.193 (**0.334**)
Has due to	0.954 / 0.941 (0.013)	0.867 / 0.773 (0.093)	0.552 / 0.243 (**0.309**)
Consists of	0.980 / 0.950 (0.030)	0.978 / 0.782 (0.196)	0.767 / 0.218 (**0.549**)
Has focus	0.955 / 0.936 (0.020)	0.843 / 0.751 (0.092)	0.492 / 0.111 (**0.381**)
Followed by	0.951 / 0.939 (0.011)	0.849 / 0.762 (0.087)	0.421 / 0.036 (**0.386**)
Occurs after	0.947 / 0.938 (0.009)	0.855 / 0.745 (0.110)	0.411 / 0.033 (**0.377**)

H Clinical Structure of Concept and Anchor Spaces

This appendix evaluates clinical structure at two levels: relations among concept embeddings and retrieval through shared semantic anchors. The two analyses use different concept universes, specified in their respective protocols; their metrics are reported separately.

H.1 Relational Structure of the Concept Embedding Space

We directly evaluate the geometry of the frozen concept representations, without fitting an additional predictor. We compare the representations actually used by the three encoders: BiomedBERT representations obtained from the OMOP concept name, MedRep representations combining textual descriptions and graph information, and the synthesis-based concept representations used by TabEHR. To avoid confounding from vocabulary coverage or fallback encoding, we restrict all analyses to the 470,484 concepts that have explicit vectors in all three representation tables. No fallback representations are included. Among these concepts, 103,848 appear in the clinical prediction datasets used in our experiments. We further restrict relation analysis to the four domains jointly covered by the three representation tables: Condition, Drug, Measurement, and Procedure.

Pairwise relation discrimination. We first test whether concepts connected by a clinical relation are closer in the embedding space than unrelated concepts. For each relation type R, we sample up to 2,000 observed relation pairs $(a, b) \in R$. For every positive pair, we construct a negative pair (a, c) by sampling c from the same domain as b, excluding concepts that are related to a. Each pair is scored using only cosine similarity between the frozen concept vectors, and AUROC is computed from the positive and negative scores. Chance performance is 0.5.

The synthesis-based representation achieves the highest AUROC for six of the seven evaluated relations, with a mean AUROC of 0.966, compared with 0.848 for MedRep and 0.736 for BiomedBERT. The only exception is *Consists of*, for which MedRep and our representation both achieve near-perfect discrimination.

To examine the source of the AUROC differences, we also report mean cosine similarities for positive and negative pairs.

Table 23: Analogy retrieval using relation-specific vector offsets. Candidate pool gives the number of concepts searched for each relation.

Relation	Candidate pool	n	BiomedBERT	MedRep	TabEHR concept space
RxNorm has ing	52,815	500	0.089	0.187	**0.439**
Contained in panel	22,652	500	0.067	0.054	**0.197**
Has due to	17,454	431	0.106	0.058	**0.174**
Consists of	52,815	500	0.251	0.191	**0.277**
Has focus	17,454	350	0.143	0.079	**0.343**
Followed by	10,927	299	0.148	0.091	**0.246**
Occurs after	10,927	278	0.086	0.211	**0.248**
Mean			0.127	0.124	**0.275**

Table 24: Detailed analogy-retrieval results. Each entry reports Hit@1 / Hit@10 / median rank.

Relation	BiomedBERT	MedRep	TabEHR concept space
RxNorm has ing	0.058 / 0.158 / 508	0.108 / 0.346 / 37	**0.348 / 0.616 / 5**
Contained in panel	0.024 / 0.154 / 304	0.032 / 0.094 / 646	**0.118 / 0.366 / 21**
Has due to	0.067 / 0.174 / 414	0.030 / 0.114 / 565	**0.111 / 0.320 / 39**
Consists of	**0.156** / 0.426 / 20	0.070 / 0.468 / 11	0.142 / **0.586 / 7**
Has focus	0.091 / 0.234 / 322	0.043 / 0.146 / 439	**0.234 / 0.523 / 8**
Followed by	0.097 / 0.241 / 180	0.047 / 0.181 / 292	**0.161 / 0.415 / 21**
Occurs after	0.043 / 0.173 / 367	0.144 / 0.320 / 62	**0.155 / 0.432 / 21**
Mean Hit@1 / Hit@10	0.077 / 0.223	0.068 / 0.238	**0.181 / 0.465**

BiomedBERT assigns high cosine similarity to both related and unrelated concept pairs, resulting in relatively small separation between the two groups. MedRep increases this separation, whereas the synthesis-based representation produces substantially larger margins across all seven relations. This suggests that the AUROC improvement reflects a more discriminative relational geometry rather than a uniform increase in cosine similarity.

Relation analogy retrieval. Pairwise proximity does not establish whether instances of the same relation share a consistent geometric structure. We therefore perform an analogy-style retrieval experiment. For two distinct pairs (a_1, b_1) and (a_2, b_2) belonging to the same relation R, we construct

$$\mathbf{t} = \mathbf{e}_{a_2} - \mathbf{e}_{a_1} + \mathbf{e}_{b_1}, \tag{23}$$

where $\mathbf{e}_c$ denotes the frozen representation of concept c. After normalizing $\mathbf{t}$, we rank all concepts in the domain of b_2 by cosine similarity to $\mathbf{t}$. We exclude a_1, b_1, and a_2 from the candidate set and record the rank of the correct target b_2. For each relation, we evaluate up to 500 such pair-of-pair examples and report mean reciprocal rank (MRR), Hit@1, Hit@10, and median rank.

Our representation achieves the highest MRR for all seven evaluated relations, with a mean MRR of 0.275, compared with 0.124 for MedRep and 0.127 for BiomedBERT. The improvement is observed across both ontology composition relations and relations spanning clinical domains.

The two evaluations probe complementary properties of the embedding space. Relation discrimination requires only that related concepts tend to be closer than unrelated concepts, whereas analogy retrieval additionally requires instances of a relation to induce consistent vector offsets. MedRep performs strongly on pairwise relation discrimination, reaching a mean AUROC of 0.848, but does not show a corresponding improvement over BiomedBERT in analogy retrieval (mean MRR 0.124 versus 0.127). In contrast, the synthesis-based representation improves both pairwise discrimination and analogy retrieval.

Notably, these evaluations use the frozen deployed representations directly: no classifier, projection, or task-specific parameters are fitted for either experiment. The results therefore characterize relational structure already present in the concept embedding space rather than predictive performance learned by a downstream model.

H.2 ANCHOR-SPACE RETRIEVAL PROTOCOL

Concept universe and panels. All four concept embedding tables cover the same $39{,}259$ concepts in the same row order. A concept is *source-observed* when it occurs in the source training records ($30{,}948$ concepts) and *target-only* otherwise ($8{,}311$); target-only concepts are present in every embedding table and in the whitening statistics but never appear in a training record (Appendix G).

Reference classes. Relevance is defined by curated terminologies: single-level CCS categories for conditions (ICD-9/10 codes reached by reversing `Maps to`, mapped through the AHRQ crosswalks), ATC level-4 classes for drugs (via `CONCEPT_ANCESTOR`), LOINC `Has component` targets for measurements (the measured analyte), and ICD-10-PCS three-character or ICD-9-Proc two-character code prefixes for procedures. Labeled concepts per domain (source / target-only): conditions $10{,}591/530$, drugs $3{,}917/3{,}698$, measurements $850/2{,}085$, procedures $13{,}522/563$. Every labeled measurement carries exactly one component. These references measure agreement with established terminology structure, not discovery of new clinical knowledge. Because ontology relations also inform concept representations, this evaluation should not be interpreted as an independent test of held-out clinical knowledge.

Representations. Each table is evaluated at three stages. *Raw*: the published embedding, ℓ_2-normalized. *Whitened*: the half-whitened, normalized representation $\check{\boldsymbol{r}}_c$, using that table's own covariance statistics. *Anchor*: the similarity vector $b(c) = (\langle \boldsymbol{a}_1, \check{\boldsymbol{r}}_c \rangle, \ldots, \langle \boldsymbol{a}_M, \check{\boldsymbol{r}}_c \rangle)$ with $M = 512$ anchors (128 per domain) built from that table by the same construction as the main model's anchors, then ℓ_2-normalized. Each table therefore receives its own whitening and its own anchors; no fitted quantity — covariance statistics or anchor vectors — is shared across tables, only the construction procedure, so the comparison measures embedding quality rather than fit to any one table's artifacts.

Retrieval and scoring. Candidates are always the labeled, source-observed concepts of the query's domain, excluding the query itself. Queries are either the candidates themselves (left panel) or the labeled target-only concepts (right panel). For a query we rank all candidates by cosine similarity in the stage representation, mark a retrieved candidate relevant when it shares at least one reference class with the query, and compute $\mathrm{NDCG@10} = \sum_{r \le 10} \mathrm{rel}_r / \log_2(r{+}1)$ divided by the same sum over an ideal ranking of the $\min(n_{\mathrm{rel}}, 10)$ relevant candidates. A query with no same-class candidate is excluded rather than scored zero; the exclusion depends only on the labels, so the scored-query set is identical for all four tables and all three stages, and every comparison is paired. Table 25 reports scored-query coverage. Measurements are scored least often because a LOINC component identifies a specific analyte and the source measurement list is a compact inpatient panel: 386 of the 850 labeled source measurements are their component's sole member, and 86% of labeled target-only measurements assay a component absent from every source measurement.

Table 25: Query coverage for anchor-space retrieval. All four embedding methods use the same scored queries. Labeled queries are scored only when at least one same-class source candidate remains after self-exclusion.

	Source-observed	Source-unobserved	
Domain	Scored	Labeled	Scored
Condition	10,585	530	530
Drug	3,871	3,698	3,585
Measurement	464	2,085	284
Procedure	13,431	563	522

H.3 SCORES AT EVERY STAGE

Table 26 reports NDCG@10 at all three stages for both panels. The orderings visible in Figure 2 already hold in the raw embeddings; relative to raw, the anchor stage changes TabEHR source-panel scores by at most -0.02 per domain and raises procedures by $+0.07$, and on the target-only panel the largest decrease is 0.03 (conditions) with procedures again gaining $+0.07$.

Table 26: NDCG@10 by embedding table, domain, representation stage, and query panel. Source-panel queries are the source-observed labeled concepts (self-excluded); target-only queries never occur in source records. Bold marks the best table per column within each domain block.

		Source queries			Target-only queries		
Domain	Table	raw	whitened	anchor	raw	whitened	anchor
Condition	TabEHR	**.692**	**.703**	**.685**	**.641**	**.651**	**.614**
	BiomedBERT	.540	.565	.534	.490	.522	.476
	MedRep	.583	.623	.518	.500	.546	.447
	MedTok	.378	.392	.361	.322	.329	.302
Drug	TabEHR	**.792**	**.807**	**.789**	**.712**	**.727**	**.710**
	BiomedBERT	.366	.405	.354	.250	.307	.243
	MedRep	.756	.771	.685	.687	.706	.587
	MedTok	.567	.600	.542	.463	.494	.446
Measurement	TabEHR	**.922**	**.924**	**.905**	**.906**	**.913**	**.889**
	BiomedBERT	.237	.325	.231	.427	.562	.387
	MedRep	.405	.465	.332	.588	.645	.465
	MedTok	.548	.570	.520	.564	.564	.558
Procedure	TabEHR	**.581**	**.615**	**.652**	**.489**	**.519**	**.560**
	BiomedBERT	**.581**	.612	.576	.452	.486	.446
	MedRep	.416	.470	.389	.335	.390	.307
	MedTok	.343	.351	.301	.269	.277	.234

H.4 A WORKED EXAMPLE

Table 27 traces one target-only query end to end: *rosuvastatin 10 MG Oral Capsule* (concept 789818), a statin never observed in the source records. Its reference class is ATC C10AA (HMG-CoA reductase inhibitors); among the 3,917 labeled source drugs, 24 share that class. Each table represents the query and all candidates by its own anchor vector $b(c)$ and ranks candidates by cosine. TabEHR retrieves ten statins (NDCG@10 $= 1.00$; MedRep and MedTok also score 1.00 on this query). BiomedBERT scores 0.13: its first eight retrievals are drugs outside the query's class — none a statin — whose most salient shared feature with the query is the surface form of the concept name (dose plus *Oral Capsule*), while statins from the source records, predominantly packaged as tablets, appear only at ranks 9 and 10. The example is a favorable one chosen for legibility (the target-only drug mean is 0.710); Table 28 adds representative and failing queries.

Table 27: Top-10 retrievals for the target-only query *rosuvastatin 10 MG Oral Capsule* at the anchor stage. ✓: shares ATC class C10AA with the query.

Rank	TabEHR		BiomedBERT	
1	rosuvastatin calcium 5 MG Oral Tablet	✓	ramipril 5 MG Oral Capsule	×
2	rosuvastatin calcium 20 MG Oral Tablet	✓	terazosin 5 MG Oral Capsule	×
3	rosuvastatin calcium 20 MG Oral Tablet [Crestor]	✓	acebutolol 200 MG Oral Capsule	×
4	rosuvastatin calcium 5 MG Oral Tablet [Crestor]	✓	niacin 250 MG ER Oral Capsule	×
5	lovastatin 20 MG Oral Tablet [Mevacor]	✓	terazosin 1 MG Oral Capsule	×
6	lovastatin 10 MG Oral Tablet	✓	niacin 500 MG ER Oral Capsule	×
7	lovastatin 20 MG Oral Tablet	✓	dutasteride 0.5 MG Oral Capsule	×
8	atorvastatin 10 MG Oral Tablet [Lipitor]	✓	disopyramide 150 MG Oral Capsule	×
9	atorvastatin 10 MG Oral Tablet	✓	lovastatin 20 MG Oral Tablet	✓
10	atorvastatin 20 MG Oral Tablet [Lipitor]	✓	lovastatin 10 MG Oral Tablet	✓
	NDCG@10	**1.00**		.13

H.5 FURTHER EXAMPLES

Table 28 lists target-only queries across all four domains, chosen to span outcomes: large-gap successes, queries near the TabEHR domain mean (representative), and failures. The failures are in-

structive. *Belzutifan* sits in ATC L01XX ("other antineoplastic agents"), a residual class collecting antineoplastics with heterogeneous mechanisms of action, so class-mates need not be pharmacologically similar; no table retrieves more than one of its 23 class-mates (best NDCG@10 = .14). *Hemoglobin urine test strip* fails for every table because the component class ignores the specimen: its same-component candidates are blood hemoglobin measurements (hemoglobin concentration and the red-cell indices MCH and MCHC), while the query lives among urinalysis concepts — a limitation of the reference classes rather than of any embedding.

Table 28: Anchor-stage NDCG@10 for further target-only queries (OMOP concept id in parentheses; n_{rel} = same-class source candidates). TabEHR target-only domain means: conditions .614, drugs .710, measurements .889, procedures .560.

Query (concept id)	Reference class	n_{rel}	TabEHR	B.BERT	MedRep	MedTok
Condition (CCS single-level)						
West syndrome (376105)	83 epilepsy; convulsions	23	**.85**	.07	.76	.00
Prim. malig. neoplasm of endocrine gland (133420)	41 cancer, other primary	47	**.64**	.41	.36	.23
Drug (ATC level 4)						
canagliflozin 300 MG Oral Tablet (43526471)	A10BK SGLT2 inhibitors	3	**1.00**	.00	.44	.23
alprazolam 2 MG Oral Tablet (781063)	N05BA benzodiazepines	28	.71	.47	**1.00**	.59
belzutifan 40 MG Oral Tablet (701701)	L01XX other antineoplastics	23	.00	.00	**.14**	.00
Measurement (LOINC component)						
Acetaminophen in Urine (3001526)	Acetaminophen	1	**1.00**	.00	.00	.00
Benzodiazepines in Urine (3000764)	Benzodiazepines	2	**.92**	.20	.48	.00
Hemoglobin in Urine by Test strip (3011397)	Hemoglobin	4	.00	.00	**.11**	.00
Procedure (ICD-PCS prefix)						
CT Scan of Bilateral Lungs (2787681)	BB2 imaging, respiratory CT	1	**1.00**	.00	**1.00**	**1.00**
Occlusion of Portal Vein, Open Appr. (2740280)	06L lower veins, occlusion	53	**.52**	.35	.22	.11

Table 29: Support-aggregation comparison (AUPRC), using the same frozen summaries and source support. Internal/external queries: MIMIC-IV validation/EHRSHOT. Pheno* averages 24 labels. **Bold**/underline: best/runner-up within each block.

	MT	LLOS	RA	UTI	Sepsis	PNA	MI	Fx	Pheno*	Mean
base rate	.048	.322	.023	.114	.082	.122	.036	.055	.120	–
Internal (MIMIC-IV val)										
Class centroid	.092	.495	.070	.361	.363	.385	.241	.192	.224	.269
k-NN vote	.323	.630	.225	.502	.492	.517	.328	.314	.324	.406
Kernel vote	.287	.608	.188	.469	.488	.518	.297	.290	.293	.382
Whitened k-NN	.336	.681	**.415**	.569	.465	.493	.356	.333	.419	.452
Linear probe (ridge)	.389	.665	.354	.578	.495	.528	.398	.385	.451	.471
TabEHR (ours)	**.515**	**.720**	.328	**.710**	**.588**	**.670**	**.494**	**.550**	**.519**	**.566**
External (EHRSHOT; MIMIC supports)										
Class centroid	.121	.360	.147	.208	.302	.246	.194	.100	.235	.213
k-NN vote	.109	.406	.150	.259	.354	.301	.223	.150	.276	.248
Kernel vote	.106	.395	.165	.249	.341	.301	.204	.143	.274	.242
Whitened k-NN	.190	.456	.115	.275	.287	.306	.175	.143	.318	.252
Linear probe (ridge)	.336	.531	.139	.369	.437	.410	.215	.200	.420	.340
TabEHR (ours)	**.496**	**.596**	**.184**	**.705**	**.684**	**.515**	**.393**	**.651**	**.507**	**.526**

I In-Context Prediction: Results and Protocols

This appendix reports diagnostic comparisons of prediction readouts and support-label perturbations, followed by their protocols and additional analyses. These diagnostics are separate from the three-seed main comparison.

I.1 Prediction behavior across institutions

We examine how the in-context predictor uses the support set. Every analysis in this subsection holds the patient summaries and the support set fixed—the same $4{,}564$ labelled MIMIC-IV training patients, disjoint from every query—and varies only the mechanism that turns support labels into a prediction.

Retrieval variants use class centroids, kernel weights, or k-nearest neighbors with and without whitening, reporting the best k per task. TabEHR leads on 17 of the 18 task–institution pairs. The internal RA exception has 34 positive queries and a paired bootstrap 95% interval of $[-.06, +.23]$ for the difference; it does not establish a reliable ranking (Appendix I.2).

The predictor clears every fixed aggregator on identical inputs. Table 29 compares TabEHR's in-context prediction against similarity-based retrieval and shallow probes that read the same frozen summaries and the same support labels. Internally, mean AUPRC rises monotonically along this ladder, from .269 (centroid vote) through .471 (ridge probe) to .566 (TabEHR): the summaries already carry neighbour structure, and the in-context predictor adds what no fixed similarity or linear readout captures. The external block sharpens the separation. With EHRSHOT queries and source labels only, retrieval stalls near the base rate (mean .21–.25), the probe recovers part of the gap (.340), and the in-context prediction retains most of its internal accuracy (.526), doubling the margin under institution shift.

The plotted internal cohort includes all 73 deceased patients and 47 survivors. Retrieval peaks at .291 AUPRC, versus .515 for in-context prediction. At $k = 64$, median $|\Delta p|$ is .317 for nearest-support label flips and .023 for random flips.

Predictions read the labels of similar support patients. The label-flip counterfactual in Figure 4b manipulates only the model input: for each query we invert the labels of its k nearest supports (cosine, summary space) and re-run inference. At k=16 over 150 queries the prediction moves by a median $|\Delta p|$ of .117, against .008 when k random labels are flipped ($14\times$; 96% of queries above the diagonal), and the effect grows monotonically in k. Survivor predictions of $p\approx$.00 rise to .45–.50

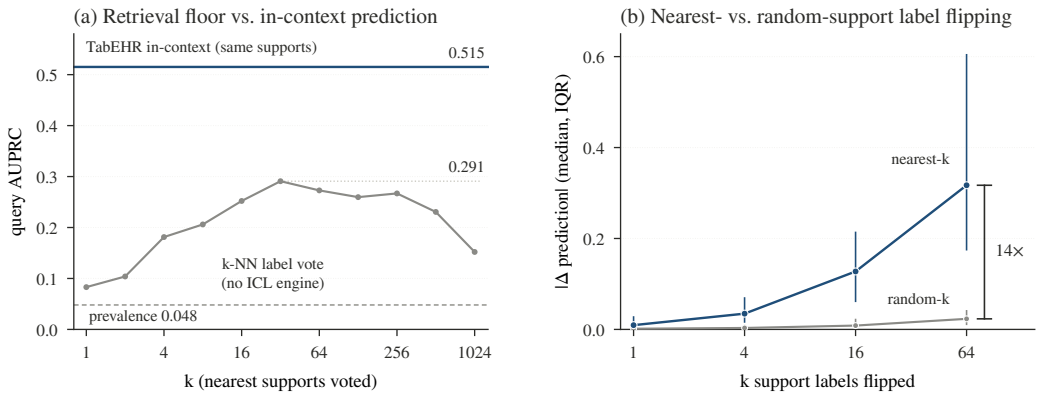

Figure 4: Support-label sensitivity on internal MT queries. (a) Retrieval and in-context prediction. (b) Nearest versus random support label flips; bars show interquartile ranges over 120 queries.

when only their 16 nearest supports are relabelled. The prediction is therefore produced by reading the labels of similar support patients, not by a support-independent decision rule.

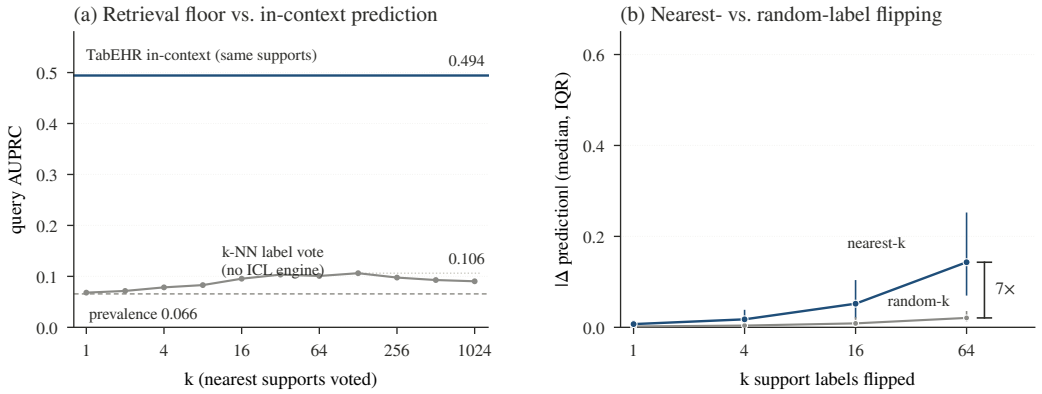

Figure 5: Support-label sensitivity on EHRSHOT MT queries with MIMIC-IV support. (a) Retrieval and in-context prediction. (b) Nearest versus random label flips, with axes matching Figure 4.

Externally, cosine retrieval reaches .106 AUPRC at prevalence .066, while in-context prediction reaches .494. Nearest-support label flips produce 3.7–6.9 times the random-control response across k, with median $|\Delta p| = .143$ at $k = 64$.

The mechanism itself transfers. Externally these two findings meet (Figure 5): cosine retrieval over the summaries collapses, yet the label-flip response of the in-context prediction persists at every k. The predictor still reads the labels of similar source patients for a target-institution query—it locates those neighbours more reliably than a fixed cosine metric does. This closes the causal chain behind the external block of Table 29: cross-institution accuracy with source labels only is delivered by similarity-matched label reading, computed in context. Per-method settings, the label-flip protocol, the RA exception, and internal support-attention measurements are provided in Appendix I.2–I.5.

I.2 Aggregation ladder: settings

All methods in Table 29 read the same frozen patient summaries: a 256-dimensional Johnson–Lindenstrauss projection of the flattened summarizer slots, extracted once from the trained checkpoint for every support and query patient. Per task and block, *class centroid* scores queries by cosine against the difference of class-mean summaries; *k-NN vote* averages the labels of the k nearest supports by cosine, $k \in \{8, 16, 32, 64, 128\}$ with the best k reported; *kernel vote* weights all supports by $\mathrm{softmax}(T \cos)$, $T \in \{30, 60, 100, 200\}$, best T reported; *whitened k-NN* first whitens the sum-

mary space with support-only statistics (shrinkage 10^{-4}); the *ridge probe* is a least-squares fit on the supports (λ=10). Reporting the best k and T favours the baselines. Pheno* repeats the full procedure per phenotype label and macro-averages the 24 labels; a label enters the average only when both classes appear among supports and among queries, matching the main evaluation. External rows use the identical MIMIC support set with EHRSHOT queries (4,403 for MT), i.e., no target labels enter any predictor.

I.3 Label-flip counterfactual: protocol

For a chosen query we invert the labels of a chosen support subset in the model input and re-run inference; nothing else changes. Each forward pass feeds the full query set and reads only the chosen query's probability, which keeps the shared sequence length fixed. Nearest supports are taken by cosine in the summary space; the random control redraws k supports per query. Internal runs use 150 queries (all 73 deceased, 77 survived) at k=16 and 120 queries (all 73 deceased, 47 survived) for the k-sweep; external runs use 120 EHRSHOT queries (60 deceased, 60 survived) with MIMIC supports, $k \in \{1, 4, 16, 64\}$. A flipped set of 16 among 4,564 supports shifts the support prevalence by at most 0.35 percentage points, which bounds the calibration component of the random control.

I.4 The RA non-win

Readmission is the one internal task where a baseline (whitened k-NN, .415) exceeds the in-context prediction (.328). The comparison rests on 34 positive queries; a paired bootstrap over queries gives a difference of $+.086$ with 95% CI $[-.062, +.234]$, so the ordering is not resolved at this sample size. RA is also structurally distinct among the nine tasks: its records are the longest (median 253 events vs. 139–173) with the largest drug share (17%), positives carry longer records than negatives (446 vs. 250 events) rather than more recent ones (median observation span 150 vs. 140 days, against 13 vs. 202 days for mortality), and its positives cohere least in the summary space (mean positive-vs-negative cosine gap .0073; a positive query's nearest support is positive in 12% of cases). Externally the in-context prediction is best on RA as well (.184).

I.5 Internal support attention corroborates the behavioural result

Reading the row-attention of query tokens over the support set (a read-only hook; the validation AUROC is reproduced exactly with the hook enabled) shows the same mechanism from inside (Figure 6). Early layers route most attention mass to the 32 global summary tokens; from the middle of the network the mass moves onto individual support patients, and deceased queries concentrate it on deceased supports (layer-10 mean .43 vs. .18 for survivors, base rate .048). The scalar "attention mass on deceased supports" alone reaches AUROC 0.944 by layer 9, against 0.954 for the full model: almost the entire prediction is legible in who the query attends to.

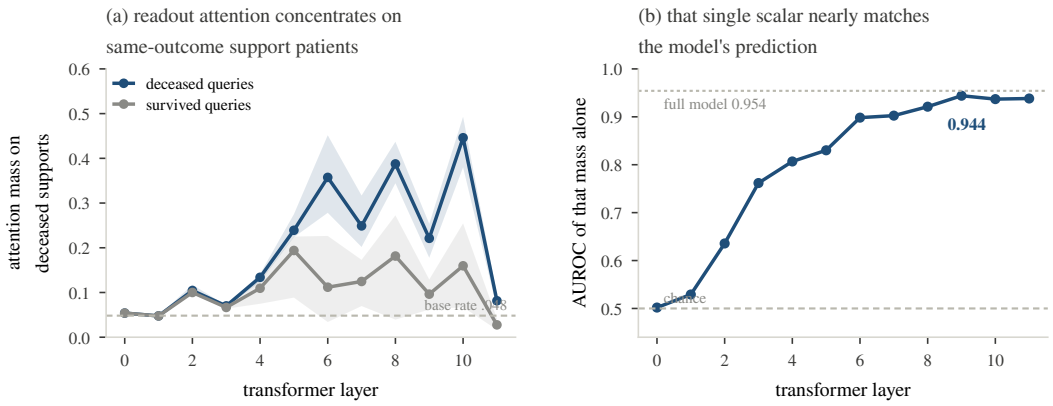

Figure 6: Internal support attention (MT, internal; 4,564 supports, 1,521 queries). (a) Median attention mass on deceased supports by layer, split by query outcome (bands: IQR). (b) AUROC of that single scalar per layer against the full model's 0.954.

J Additional Analyses of Patient Representations

These analyses complement the nine-task comparison with training-seed variation, statistical-input comparisons, and diagnostics of the learned representation. Checkpoint-specific analyses are identified separately from the three-seed main results.

J.1 Task-level transfer and training-seed variation

We compare TabEHR with the two MEDS-Tab predictors on the same nine external tasks. The largest improvements occur for Sepsis, MI, and Fx, whereas MT, UTI, and PNA favor both statistical comparators. The macro-average therefore describes an aggregate advantage rather than uniform improvement across outcomes.

Table 30: Observed external differences computed from full-precision saved aggregate metrics. TabEHR uses the three-seed mean; comparators use the saved MEDS-Tab metrics reported in the main table. No statistical significance is asserted.

Task	XGB ΔROC	XGB ΔPRC	TFM ΔROC	TFM ΔPRC
MT	-0.018	-0.100	-0.018	-0.109
LLOS	+0.002	+0.001	-0.011	-0.032
RA	-0.007	+0.005	+0.045	+0.046
Pheno	+0.028	+0.035	+0.043	+0.044
UTI	-0.008	-0.015	-0.010	-0.025
Sepsis	+0.100	+0.196	+0.145	+0.214
PNA	-0.026	-0.066	-0.015	-0.100
MI	+0.142	+0.158	+0.126	+0.182
Fx	+0.225	+0.498	+0.225	+0.483
Macro	+0.049	+0.079	+0.059	+0.078

Table 31: TabEHR across three source-selected training seeds. Mean ± SD reports between-seed variation.

	MIMIC-IV		EHRSHOT	
Run	ROC	PRC	ROC	PRC
Seed 100, step 6,000	.900	.562	.811	.525
Seed 200, step 7,000	.896	.554	.787	.479
Seed 300, step 7,000	.901	.559	.816	.526
Mean ± SD	$.899 \pm .003$	$.558 \pm .004$	$.804 \pm .015$	$.510 \pm .027$

Each TabEHR run exceeds both reported statistical comparators in external macro AUROC and AUPRC. The between-seed SDs are .015 and .027, respectively. The task-level differences are descriptive; patient-paired confidence intervals have not been computed and are not inferred from rounded aggregate metrics.

J.2 Value processing and statistical feature construction

The main statistical baselines use raw-value MEDS-Tab summaries. We also evaluate source-fitted value processing with the same six aggregations, and the processed values with a custom statistical schema. Each input table uses 512 selected columns, with feature ranking fitted separately on the source. Value processing jointly changes unit conversion, range handling, normalization, and duplicate consolidation; it is not a normalization-only intervention.

For the TFM, value processing changes external macro AUROC/AUPRC from $.746/.432$ to $.751/.441$, and the custom schema reaches $.759/.444$. These changes do not close the gap to TabEHR ($.804/.510$). The comparison supports the complete learned representation rather than isolating anchor aggregation from input processing or representation capacity.

Table 32: Statistical-input comparisons with 512 columns across nine tasks. Int./Ext.: MIMIC-IV/EHRSHOT.

Input table	Predictor	Int. ROC	Int. PRC	Ext. ROC	Ext. PRC
Raw / default six	XGBoost	.885	.540	.756	.431
Raw / default six	TFM	.898	.549	.746	.432
Processed / default six	XGBoost	.878	.533	.756	.433
Processed / default six	TFM	.894	.553	.751	.441
Processed / custom	XGBoost	.887	.550	.758	.435
Processed / custom	TFM	.893	.562	.759	.444

J.3 Attention allocation and patient-specific readouts

We examine the seed-100 main checkpoint at step 6,000. Attention assigns about 99% of its mass to real events, with similar aggregate allocation on MIMIC-IV and EHRSHOT. More than half of the attention mass crosses the anchor's construction domain. Thus, a domain used to construct an anchor does not restrict which clinical events that anchor can read.

Table 33: Attention and readout diagnostics for the seed-100 model at step 6,000. Definitions and query sampling are described below.

Measurement	MIMIC-IV	EHRSHOT
Mean real-event attention $1 - \alpha_{m,*}$	.989	.991
Normalized attention entropy over valid events	.860	.878
Attention mass crossing anchor-construction domains	.622	.604
Between-patient readout variation per slot	.372	.314
Redundancy between slot readouts	.801	.743

Diagnostics are weighted by query-record count across nine tasks, using full MIMIC-IV query sets and fixed 2,000-record EHRSHOT subsamples per task. Real-event attention complements mean null-token attention, and entropy is normalized by the log number of valid candidates. Cross-domain attention counts mass outside an anchor's construction domain. Variation is the median across slots of the between-patient readout standard deviation relative to mean norm; redundancy is the mean absolute correlation between patient-centered slot readouts. These are learned-attention diagnostics, not hard nearest-anchor construction counts.

Readouts vary across patients, but their high between-slot correlation indicates substantial shared information. These diagnostics describe the learned representation; they do not establish that each slot is an independent clinical factor or a disease detector.

J.4 SENSITIVITY TO THE SOURCE SUPPORT BUDGET

We vary only the number of MIMIC-IV support patients while retaining the same EHRSHOT queries and the seed-100 model at step $4{,}000$. Five nested, stratified support draws are used for each reduced budget, independently of query outcomes. This fixed-checkpoint sensitivity analysis is distinct from the source-selected three-seed main comparison and does not introduce target labels.

Table 34: EHRSHOT performance by MIMIC-IV support budget, using seed 100 at step $4{,}000$. Values are macro means $\pm$ SD across five support draws; full support uses one draw.

Source support budget	Ext. ROC $\pm$ SD	Ext. PRC $\pm$ SD
128 patients	$.677 \pm .012$	$.283 \pm .012$
256 patients	$.729 \pm .006$	$.363 \pm .024$
512 patients	$.758 \pm .009$	$.438 \pm .006$
1,024 patients	$.789 \pm .006$	$.480 \pm .007$
2,048 patients	$.812 \pm .005$	$.517 \pm .006$
4,096 patients	$.815 \pm .003$	$.526 \pm .007$
Full eligible source training set	.815	.530

The full eligible support contains $4{,}267$–$13{,}257$ patients per task. At that budget, the reconstruction path agrees with default evaluation within .002 AUROC/AUPRC. The seed and checkpoint differ from the three-seed main result; no target-label budget is varied.

Performance increases with the source support budget, with gains becoming smaller beyond $2{,}048$ patients. Variation across support draws also decreases. This pattern concerns the source context supplied at inference, not few-shot learning from target-institution labels.

Table 35: Support-size comparison on seven outcomes not explicitly supervised by the source tasks. Macro AUROC at $k = 16$ and 128 training labels per class, averaged over five support replicates.

Representation	Predictor	$k = 16$	$k = 128$
No target-institution pretraining			
Count	GBM	.600	.685
	LR	.630	.703
	TFM	.644	.722
MedTok + BEHRT	LR	.564	.605
	TFM	.565	.628
MedRep + BEHRT	LR	.573	.617
	TFM	.567	.614
TabEHR (ours)	LR	.637	.747
	TFM	**.684**	**.782**
With target-institution pretraining			
CLMBR-T-base	LR	.696	.764
	TFM	.675	.747

K Few-Shot Evaluation (Provisional)

This appendix supplements section 4.3 with protocol and implementation details, support-size comparisons, and CLMBR-T-base results. The current evaluation uses one source-training checkpoint.

K.1 Reusing source-supervised representations on target tasks

Table 35 extends the main comparison to $k = 128$ on the seven outcomes not explicitly supervised by the source tasks. The same MIMIC-trained representation and frozen TFM are used at both support sizes. Sampling and label budgets are defined in Appendix K.2.

TabEHR + TFM achieves AUROC of .684 and .782 at $k = 16$ and 128, compared with .644 and .722 for Count + TFM. CLMBR-T-base + LR reaches .696 and .764, respectively. The relative ordering therefore depends on support size and the representation–predictor combination; differences in supervision and input processing are detailed below.

K.2 Official EHRSHOT few-shot benchmark

We use the official fifteen-task EHRSHOT benchmark, patient partitions, and support replicates (Wornow et al., 2023), separately from the reconstructed nine-task cohorts. Target training examples provide support for disjoint test patients without updating TabEHR parameters. Support centering excludes query patients, and source-fitted numerical preprocessing remains fixed. Events are restricted to each instance's prediction time (Appendix C.4).

Benchmark tasks and label budgets. For each binary task, k labels per class provide $2k$ training labels. TFM rows use only these labels as context; fitted heads additionally use $2k$ validation labels for model selection. Training-shot counts, but not total label budgets, are matched. CXR samples support separately for each of fourteen findings. Counts refer to labeled instances, not necessarily distinct patients. Finite-shot scores average five support replicates; All, used only in the reproduction check, uses the complete training split and one replicate. Patient-bootstrap intervals have not been computed.

Released label and sampling conventions. Shot indices address label rows after filtering to the designated split and sorting by patient identifier. The five laboratory tasks binarize their four-level severity labels as any abnormality. The chest-radiograph task uses fourteen independent findings encoded as a bit mask, whose bit order is the reverse of the listed finding order. These conventions and the reproduced baseline scores are checked in Appendix K.5.

AUROC is averaged over findings within each task, then over tasks and support replicates. CXR contributes one task to the fifteen-task mean. This differs from the fourteen binary-task average in Context Clues (Wornow et al., 2025).

K.3 SOURCE–TARGET OUTCOME CORRESPONDENCE

Groups are defined relative to TabEHR's source label definitions, including the 24 labels within PHENO, rather than task names alone (Table 36). Source phenotyping labels conditions present during admission, while target new-diagnosis tasks concern first occurrence within a year and exclude patients with prior diagnoses. Related conditions therefore need not define the same prediction task.

Table 36: Correspondence between the fifteen benchmark tasks and our source labels, used for the source-overlap analysis. Source labels prefixed PHENO are members of the 24-label phenotype set. Target definitions follow the published benchmark description; the labelers themselves are not part of the released assets, so the timing and eligibility columns are cited rather than measured.

Benchmark task	Related source label	How the label definition differs	Group
`guo_los`	LLOS	same endpoint and prediction time	same label
`guo_readmission`	RA	same endpoint and horizon	same label
`new_acutemi`	MI; PHENO acute myocardial infarction	source labels presence in the admission; target labels first onset within a year, incident patients only	related outcome
`new_hypertension`	PHENO essential hypertension; hypertension with complications	as above	related outcome
`new_hyperlipidemia`	PHENO disorders of lipid metabolism	as above	related outcome
`lab_hyperkalemia`	PHENO fluid and electrolyte disorders	source labels a diagnosis in the admission; target labels the severity of a specific assay at order time	related outcome
`lab_hyponatremia`	PHENO fluid and electrolyte disorders	as above	related outcome
`guo_icu`	—	no source label for intensive-care transfer	unseen outcome
`new_celiac`	—	no source label for coeliac disease	unseen outcome
`new_lupus`	—	no source label for lupus	unseen outcome
`new_pancan`	—	no source label for pancreatic cancer	unseen outcome
`lab_anemia`	—	no source label for anaemia; gastrointestinal haemorrhage is a cause, not the condition	unseen outcome
`lab_thrombocytopenia`	—	no source label for thrombocytopenia	unseen outcome
`lab_hypoglycemia`	(PHENO diabetes, indirect)	the source labels the diabetes diagnosis, not a hypoglycaemic event	unseen outcome
`chexpert`	PHENO pneumonia; pleurisy, pneumothorax, pulmonary collapse; other lower respiratory disease; respiratory failure	six of the fourteen findings name conditions the source supervises, eight do not; the label is read from a radiology report rather than from diagnosis codes	mixed, reported separately

The groups contain two same-label tasks, five related outcomes, and seven outcomes not explicitly supervised at the source; CXR is mixed and reported separately. The seven-outcome mean at $k = 16$ is the designated primary endpoint. "Unseen" refers to TabEHR's source labels, not the absence of related concepts in source records or in other models' pretraining.

K.4 COMPARATOR IMPLEMENTATIONS

All TFM rows use the same frozen backbone. Count and CLMBR-T-base features come from the benchmark release; fitted predictors are refitted and checked against released scores (Appendix K.5). TabEHR uses source clinical supervision, whereas the BEHRT representations come from self-supervised source pretraining. CLMBR additionally uses pretraining at the target institution. These comparisons do not isolate aggregation architecture from supervision or pretraining access.

Checkpoint and replication. The current TabEHR results use the seed-100, step-4,000 checkpoint. They are distinct from the three-seed, source-validation-selected results in the nine-task comparison. The five few-shot replicates vary the released support samples, not the source-training seed. A final matched comparison across source-training seeds remains to be completed.

Count inputs and reporting conventions. Count + TFM reads 512 of the 52,025 count columns, ranked by non-zero frequency in the k-shot support set, and scales them to unit maximum absolute value fitted on that same support set. Neither step reads a target label or a query row. The logistic and boosted rows use all 52,025 columns, so that row trades column coverage for the predictor's column budget. Count + GBM uses XGBoost here; the released boosted head uses LightGBM. Reproduced logistic-head scores and implementation deviations are documented in Appendix K.5.

Representations from EHR sequence models. MedTok + BEHRT and MedRep + BEHRT use frozen, mean-pooled representations from MIMIC-IV masked-language-model checkpoints, without target updates. They do not use the task-finetuned checkpoints in Table 1. Events outside the source vocabulary become unknown tokens: 42.8% overall, comprising 48.9% of measurements, 29.4% of drugs, 23.0% of procedures, and 8.2% of conditions.

TabEHR with a fitted head. TabEHR + LR reads a 2048-dimensional Gaussian random projection of the flattened anchor slots, concatenated with the slot mean. The projection uses a fixed seed, so every row and every task sees the same map, and the scaler and the logistic head are fitted on the support rows alone. TabEHR + TFM reads the slots in full, so that row pair carries this compression in addition to the change of head.

K.5 REPRODUCTION OF THE RELEASED BASELINES

We refit predictors on the released count and CLMBR-T-base features using the official splits and support samples. Label joins, class balance, prediction cutoffs, laboratory binarization, CXR bit order, and metric aggregation follow the conventions above. Logistic-head scores differ from the released values by at most .0024 AUROC (Table 37). Count-feature scaling is required for optimizer convergence. The boosted comparison uses XGBoost here and LightGBM in the release, so it is not an exact implementation reproduction.

Table 37: Reproduction of the released EHRSHOT baselines. Macro AUROC over the fifteen tasks, aggregated as the official code aggregates: over sub-tasks, then tasks, then shot replicates. "Released" is the number in the distributed result files. The released boosted head is LightGBM; ours is XGBoost, so this row compares two implementations of the same method rather than two runs of one implementation.

Method	$k = 16$	$k = 128$	All
CLMBR-T-base + LR, released	.661	.732	.772
CLMBR-T-base + LR, ours	.661	.732	.772
Count + LR, released	.606	.672	.709
Count + LR, ours	.603	.674	.710
Count + GBM, released	.573	.692	.729
Count + GBM, ours	.590	.662	.692
Deviation, logistic heads	.0024	.0012	.0010
Deviation, boosted head	.0162	.0303	.0369

K.6 ADDITIONAL COMPARISON WITH CLMBR-T-BASE

Table 38 supplements the main comparison with CLMBR-T-base, which uses target-institution pretraining. The TabEHR row is repeated as a reference; the other methods are reported in Table 2. The methods differ in source supervision and pretraining access, and the LR comparison additionally differs in predictor and target validation-label use. These contrasts therefore do not isolate an architecture effect.

Table 38: Additional CLMBR-T-base comparison at $k = 16$. Macro AUROC averaged over five support replicates. Only CLMBR uses target-institution pretraining. CXR averages fourteen findings.

Representation	Predictor	Overall (15)	Same (2)	Related (5)	Unseen (7)	CXR (14 findings)
TabEHR	TFM	.659	.672	.634	.684	.581
CLMBR-T-base	LR	.661	.687	.611	.696	.622
	TFM	.640	.646	.596	.675	.600