SparseEHR: Scalable Foundation Modeling for Structured EHR via Conditional Computation

Meysam Ghaffari [1] **Animesh Agarwal** [1] **Nina Fatehi** [1] **Lambert Leong** [1] **Thomas Linden** [1]
Reihaneh Hassanzadeh [1] **Carlos Morato** [1]

Abstract

Structured electronic health records (EHRs) are a natural substrate for healthcare foundation models, but dense transformers remain expensive to scale across heterogeneous code vocabularies and long patient histories. We present **SparseEHR**, a hybrid dense-to-sparse transformer for structured EHR sequences that uses dense warm-start layers followed by mixture-of-experts (MoE) layers that activate one router-selected expert plus an always-active shared expert for each token. SparseEHR is pretrained on longitudinal diagnosis and procedure sequences from approximately 50 million de-identified individuals in the OptumLabs Data Warehouse. In zero-shot transfer to MIMIC-IV, without fine-tuning, SparseEHR achieves 0.463 Recall@10 and 0.551 Recall@20 for next-visit ICD-10 prediction under our fixed preprocessing and code-mapping protocol; published baselines are included for context rather than as a protocol-matched state-of-the-art claim. We report implementation-level parameter accounting that separates stored capacity from active per-token computation across dense FFNs, shared experts, routed experts, attention, embeddings, and routers. These results frame SparseEHR as a structured-EHR capacity and transfer trade-off, not as a universal serving-speed improvement.

1. Introduction

Structured health data lies at the center of clinical machine learning: diagnoses, procedures, medications, labs, and physiologic measurements define longitudinal patient trajectories and support forecasting, risk stratification, and operational decision-making. A particularly important clinical problem is next-event prediction from large longitudinal EHR, where anticipating future diagnoses can support earlier risk stratification, care planning, and population-level resource allocation, but existing dense models become costly as patient histories and medical vocabularies grow. Recent transformer models for structured EHR, including BEHRT, Med-BERT, CEHR-BERT, and more recent large-scale pretrained EHR models (Tang et al., 2023; Wornow et al., 2024), have improved representation learning and downstream prediction, but scaling dense models remains expensive when sequences are long, code vocabularies are large, and pretraining corpora span tens of millions of patients (Li et al., 2020; Rasmy et al., 2021; Pang et al., 2021; Du et al., 2022; Jiang et al., 2024b).

Three properties of structured EHR make this scaling problem particularly acute. First, timelines are temporally clustered into visits and day-level bundles rather than naturally uniform sequences. Second, the token space is *heterogeneous*: common chronic-disease backbone codes coexist with rare diagnoses, procedures, and workflow markers. Third, use at population scale is constrained by *compute*: a practically useful foundation model must serve long histories and large cohorts while making explicit trade-offs among stored capacity, active computation, memory, and latency.

Mixture-of-experts (MoE) architectures offer a compelling alternative because they decouple stored model capacity from the subset of expert parameters activated for a token (Shazeer et al., 2017; Fedus et al., 2022). However, naive sparse routing can be unstable, fragment shared structure, and behave poorly on heterogeneous health data where frequent and rare clinical concepts coexist. Recent medical MoE work has largely focused on medical vision-language or multimodal imaging tasks (Jiang et al., 2024a; Chopra et al., 2025); here we study sparse routing for longitudinal structured diagnosis/procedure-code trajectories. In a workshop paper, the key question is therefore simple: *can conditional computation provide useful capacity and transfer trade-offs for **structured-EHR** foundation models?*

We study this question with **SparseEHR**, a hybrid transformer for structured EHR sequences. The model uses

[1]Optum AI, UnitedHealth Group, Minneapolis, Minnesota, USA. Correspondence to: Carlos Morato <carlos.morato@optum.com>.
Proceedings of the Workshop on Structured Data for Health at the 43rd International Conference on Machine Learning, Seoul, South Korea. Copyright 2026 by the author(s).

dense layers early to learn shared patient-state representations, then activates sparse experts later to specialize on heterogeneous medical code patterns. This short version intentionally focuses on one clean message and two pieces of evidence: **(i)** zero-shot cross-system transfer from OptumLabs Data Warehouse (OLDW) pretraining to MIMIC-IV evaluation, and **(ii)** implementation-level accounting that distinguishes stored model capacity from active per-token computation. Together, these results position SparseEHR as a candidate approach for structured health data modeling, while keeping efficiency and state-of-the-art claims appropriately bounded.

2. SparseEHR for Structured EHR

Input representation. Each patient is represented as a chronological sequence of typed tokens built from demographics and structured clinical codes. We prefix the sequence with static attributes (e.g., age bucket, sex, region), then append daily bundles of standardized diagnosis and procedure tokens with explicit delimiters. ICD codes are normalized to canonical prefixes and CPT codes to fixed numeric forms, producing a compact structured vocabulary while retaining clinically meaningful granularity.

Task formulation. Given a patient timeline $x = (x_1, \ldots, x_t)$, SparseEHR is trained to predict the next structured token autoregressively:

$$p(x_{t+1} \mid x_{\leq t}) = \mathrm{softmax}(Wh_t). \tag{1}$$

The vocabulary projection is weight-tied to the input token embedding matrix, so the language-model head does not add an independent vocabulary-sized parameter matrix. At evaluation time, we roll the model forward to generate the next visit bundle token by token and score whether the true ICD-10 codes appear within the top-K predictions. This formulation lets one pretrained model support both representation learning and next-event forecasting.

Hybrid dense–MoE architecture. SparseEHR is a 16-layer transformer with residual width d_{model}=2048 and a vocabulary of 20,656 structured code tokens. Attention is computed in a projected 2040-dimensional subspace with 12 heads of dimension 170: Q/K/V projections map from 2048 to 2040, and the attention output projection maps back to 2048. The first D=4 layers are standard dense transformer blocks. The remaining $L - D$=12 layers are MoE blocks with four router-selectable experts plus one always-active shared expert. For token representation h_i, the router produces logits over the four routed experts, selects the top k=1 expert, and applies softmax only over the selected logits:

$$\ell_i = W_r h_i, \qquad S_i = \mathrm{TopK}(\ell_i, k), \qquad k = 1, \tag{2}$$

$$\alpha_{ij} = \frac{\exp(\ell_{ij})}{\sum_{m \in S_i} \exp(\ell_{im})}, \qquad j \in S_i. \tag{3}$$

The MoE output combines the always-active shared expert with the selected routed expert:

$$y_i = \mathrm{Expert}_{\mathrm{shared}}(h_i) + \sum_{j \in S_i} \alpha_{ij}\, \mathrm{Expert}_j(h_i). \tag{4}$$

Thus each MoE layer activates one routed expert and one shared expert for each token. Inactive routed experts are counted as stored capacity but not as active expert computation. Throughout the paper, *stored parameters* include all experts and all non-expert transformer components, while *active parameters/token* count only the components executed for a token: attention/projection/norm components, dense-prefix FFNs, the shared expert in each MoE layer, the selected routed expert, and router operations. Appendix B gives the component-level accounting used to distinguish stored and active parameters.

Why conditional computation is a natural fit for structured EHR. Structured EHR is not uniformly complex from token to token. Much of a patient timeline is composed of common population-level backbone patterns—recurrent chronic diagnoses, routine outpatient procedures, and visit-structure markers—that benefit from shared dense processing. In contrast, rarer clinical trajectories, specialty-specific procedures, and context-dependent code combinations are more heterogeneous and may benefit from extra stored capacity. A hybrid dense–MoE split therefore matches the data-generating structure of EHR: dense warm-start layers learn stable global patient-state representations, while later experts spend capacity on tokens whose clinical context is more specialized. This design increases stored capacity while activating only a subset of routed experts for each token; it should not be interpreted as faster or cheaper than every dense alternative.

Why the hybrid split matters in practice. Dense warm-start layers absorb high-frequency global structure before routing begins, which stabilizes sparse training on temporally clustered, heterogeneous patient histories. The later MoE layers then provide conditional capacity for diverse clinical code interactions at scale. The main paper therefore treats the hybrid design as the selected SparseEHR configuration and reports audited capacity/compute accounting against both an all-MoE reference and an all-dense control.

Training stability mechanisms. Conditional computation only helps if routing remains stable on long-tailed clinical data. In addition to the shared expert pathway, SparseEHR uses grouped-query attention (Ainslie et al., 2023), rotary positional embeddings (Su et al., 2021), and an expert-utilization term/monitor that penalizes extreme

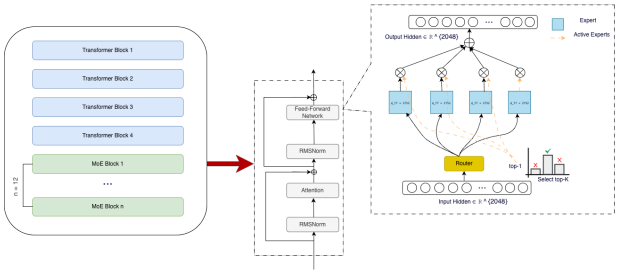

Figure 1. A compact view of the SparseEHR design principle: shared early processing, one shared expert plus one routed expert in later MoE layers, and transfer through a common structured tokenization scheme.

concentration. Because this term is based on observed routing behavior rather than proposed as a new differentiable load-balancing algorithm, we treat load balancing as an implementation detail rather than a methodological contribution.

3. Experimental Setup

Pretraining corpus. We pretrain SparseEHR on longitudinal structured records from approximately 50 million de-identified individuals in OLDW (Theel et al., 2015). Training uses an autoregressive next-code objective over chronological diagnosis and procedure sequences. Sequences are truncated or padded to length 2048 and optimized with a causal next-token loss.

Primary transfer evaluation. To isolate cross-system transfer, we evaluate *without fine-tuning* on MIMIC-IV next-visit ICD-10 prediction (Johnson et al., 2023). MIMIC records are converted into the same structured format used for pretraining: admissions are ordered chronologically, diagnosis and procedure codes are standardized, and the resulting timeline is fed directly to SparseEHR. The evaluation is deliberately *code-based rather than note-based*; no clinical notes or MIMIC labels are used for adaptation. We report Recall@K, which is standard for multi-label diagnosis prediction with large sparse output spaces.

Data harmonization for zero-shot transfer. Because MIMIC-IV contains a mix of coding systems, we apply a fixed preprocessing pipeline before evaluation. Raw patient, admission, diagnosis, and procedure tables are first assembled into temporally ordered patient timelines. Legacy diagnosis codes are then mapped into the ICD-10-oriented token space used by SparseEHR, and procedures are converted to the canonical procedure format used during pretraining. Crucially, this harmonization step is frozen before evaluation and does not use MIMIC labels for tuning. The goal is not to optimize a benchmark-specific adapter, but to test whether a large structured-EHR model can transfer through a common schema alone. Because cohort construction, preprocessing, code conversion, and Recall@K definitions may differ across prior reports, we treat published baselines as contextual rather than protocol-matched comparisons.

Baselines and efficiency analysis. For transfer, we compare against representative public baselines spanning structured-EHR pretraining and recent LLM-based diagnosis models: MedBERT, MERA, NECHO, and recent foundation-model-based EHR approaches (Rasmy et al., 2021; Ma et al., 2025; Koo, 2024; Anonymous, 2025). For capacity and compute, we follow the implementation audit by distinguishing shared experts, routed experts, dense FFNs, attention/projection weights, embeddings, routers, stored parameters, and active parameters/token. We avoid a blanket E/k compute-reduction claim, because top-k routing activates only a subset of routed expert capacity and is not automatically cheaper than a standard dense Transformer FFN.

Implementation details. SparseEHR is trained with a causal next-token objective over sequences of maximum length 2048. Training uses causal attention masking, with padding and non-target prompt tokens excluded from the loss where appropriate. The attention stack uses grouped-query attention and rotary positional embeddings, while the feed-forward stack follows the dense-to-sparse split described above. For transfer, we evaluate bundle prediction autoregressively: given a patient context, the model generates the next visit token by token and we score whether the ground-truth ICD-10 codes appear in the top-K predictions at each step. This setup keeps the structured prediction task aligned across datasets and avoids any MIMIC-specific calibration.

4. Results

Primary finding. SparseEHR shows promising zero-shot ICD prediction on MIMIC-IV while making explicit the distinction between stored expert capacity and active per-token computation.

Zero-shot transfer to a new health system. SparseEHR achieves 0.463 Recall@10 and 0.551 Recall@20 on MIMIC-IV in a zero-shot setting (Table 1). These values are higher than the published baselines we include for context, but the comparison should not be read as a protocol-matched state-of-the-art claim because cohort construction, mapping, and metric definitions may differ. The result instead suggests that large-scale structured-EHR pretraining can transfer across institutions when records are expressed in a common typed-token schema.

Table 1. Zero-shot transfer from OLDW pretraining to MIMIC-IV next-visit ICD-10 prediction. SparseEHR is evaluated without fine-tuning or MIMIC-specific label adaptation. Published baselines are included for context and are not fully protocol-matched because preprocessing, cohort construction, code mapping, and metric definitions may differ. Capacity/compute accounting definitions are reported in Appendix B.

Method	R@10	R@20
MedBERT	0.259	0.338
MERA (BioMistral-7B)	0.396	0.491
NECHOv3 (GPT-4o-mini)	0.363	—
SparseEHR	**0.463**	**0.551**

Capacity accounting for conditional computation. The hybrid design increases stored expert capacity while limiting each token to the always-active shared expert and a selected routed expert in each MoE layer. Following the implementation audit, we define stored parameters as all model parameters and active parameters/token as only the parameters executed for a token. This accounting is more precise than the common but misleading E/k shorthand: sparse routing activates a subset of routed expert capacity, but the shared expert and non-expert Transformer components are still always active. We therefore frame SparseEHR as a capacity and transfer trade-off rather than a universal deployment-efficiency improvement.

Transfer is not just benchmark adaptation. The zero-shot setup is important substantively, not only procedurally. Many high-performing diagnosis predictors depend on dataset-specific supervision, task-specific prompt engineering, or multimodal components tuned on the target institution. SparseEHR instead transfers through the structured code space itself. Once MIMIC-IV is expressed in the same typed-token format, the pretrained model can forecast the next visit without seeing any MIMIC labels during optimization. This makes the result especially relevant for structured health data research, where schema alignment and transferable representation learning are often more practical than repeated end-to-end retraining at each site.

Supportive diagnostics and interpretation. Qualitative token inspection suggests *partial specialization by EHR function*: some experts absorb diagnosis-backbone tokens, while others focus on outpatient workflow, procedure-linked transitions, or context-specific chronic care. Rare-code prediction remains harder than common-code prediction, but recall still rises with larger K, indicating that sparse experts retain signal for infrequent concepts. Together, these results suggest that conditional computation is not merely a generic scaling trick, but a useful inductive bias for heterogeneous, long-tailed clinical timelines. The appendix reports the fuller routing and ablation analyses.

Secondary in-distribution validation. On a secondary OLDW incident-diabetes endpoint, SparseEHR is competitive with structured-EHR baselines and stronger than prompted GPT references under a shared hard endpoint. XGBoost is the strongest classical baseline we added after review, with AUROC 0.752, AUPRC 0.721, and F1 0.732, close to SparseEHR's thresholded F1 of 0.735 (Appendix Table 5). GPT-5 and GPT-4.1 are therefore reported only as secondary LLM references rather than as primary structured-EHR baselines (Appendix Table 6). Because the cohort is small and the decision rule is validation-selected, we treat this endpoint as supportive rather than definitive superiority evidence.

5. Implications for Structured Health Data

Transfer through schema, not site-specific retraining. A practical challenge in structured-health modeling is that many institutions cannot afford to retrain a large model each time the local schema, outcome definition, or patient population changes. The SparseEHR transfer result points to a different strategy: invest in large-scale pretraining over a stable typed-token schema, then adapt new sites primarily through harmonization into that schema rather than through full supervised re-optimization. This is especially attractive for settings where diagnosis and procedure data are abundant but gold labels for a new downstream task are scarce or delayed.

Selective computation matches the structure of EHR. The routing summary in Table 2 suggests that sparse experts in structured EHR do not simply memorize narrow disease labels. Instead, they tend to organize around broader *functions of record generation*: common diagnostic backbone, ambulatory workflow, procedure-linked transitions, and context-specific chronic care. That is a good fit for real EHR, where the same disease token can appear in different operational contexts and where rare but important events are embedded inside common recurrent structure. A dense prefix plus sparse suffix therefore acts as both a systems design and a representational bias: shared early layers capture population-common context, while later experts spend extra capacity where heterogeneity is highest.

Table 2. Qualitative routing patterns observed in the selected $D{=}4$ model.

Layer	Representative specialization pattern
4	Earliest routing split between diagnosis-backbone tokens and workflow-linked procedures, demographics, and visit-structure markers.
6	Sharper separation into cardiometabolic follow-up, broad chronic-diagnosis backbone, musculoskeletal/skin care, and mixed screening-workflow context.
11	Shared high-frequency diagnoses remain distributed across experts, while residual experts capture preventive care, procedure-linked transitions, and lower-frequency context.
15	Most mature specialization by EHR function: common diagnosis backbone, visit transitions, chronic ambulatory patterns, and outpatient management/lab-office workflow.

6. Discussion and Limitations

This workshop version prioritizes two signals: zero-shot transfer under a fixed structured-code protocol and implementation-level stored-versus-active parameter accounting. Our study is limited to structured diagnosis/procedure sequences plus demographics, and does not yet evaluate multimodal fusion, fairness, calibration, prospective deployment, or broader external validation beyond MIMIC-IV. MIMIC-IV results are also limited by a small retained cohort after harmonization, code-conversion noise, and non-protocol-matched published baselines. The model uses chronological ordering and day-level bundles but does not explicitly encode continuous time gaps or time-decay effects; explicit temporal encodings remain future work.

The main methodological takeaway is that structured EHR need not use the same expert capacity for every token. Common events benefit from shared representations, while rarer or workflow-specific events can benefit from sparse specialization. SparseEHR suggests that a dense prefix plus shared-and-routed expert layers is a plausible fit to this structure, but dense controls remain essential for interpreting any efficiency claim. Future work should test whether this extends to medications, labs, time-series signals, and multimodal records (Anonymous, 2024).

7. Conclusion

We presented SparseEHR, a structured-EHR foundation model based on conditional computation. A hybrid dense–MoE transformer pretrained on 50M de-identified longitudinal records transfers zero-shot to MIMIC-IV under a fixed code-mapping protocol while using shared-and-routed expert layers to separate stored expert capacity from active per-token computation. These results support selective computation as a promising capacity/transfer trade-off for population-scale structured health modeling.

Impact Statement

This paper presents work whose goal is to advance structured-data modeling for healthcare. Potential societal consequences include improved representation learning for clinical prediction and population-scale forecasting, but responsible use requires careful external validation, calibration, privacy safeguards, fairness assessment, and clinical oversight before any deployment. The results should therefore be interpreted as evidence for a capacity and transfer trade-off in structured EHR modeling rather than as a claim of clinical readiness.

References

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. GQA: Training generalized multi-query transformer models from multi-head checkpoints. *arXiv preprint arXiv:2305.13245*, 2023.

Shivang Chopra, Lingchao Mao, Gabriela Sanchez-Rodriguez, Andrew J. Feola, Jing Li, and Zsolt Kira. MedMoE: Modality-specialized mixture of experts for medical vision-language understanding. *arXiv preprint arXiv:2506.08356*, 2025.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

Alistair E. W. Johnson, Lucas Bulgarelli, Leo Shen, Abigail Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Sophie Hao, Benjamin Moody, Brian Gow, Li-wei H. Lehman, Leo Anthony Celi, and Roger G. Mark. MIMIC-IV, a freely accessible electronic health record dataset. *Scientific Data*, 10(1):1, 2023.

Hyeongu Koo. Next visit diagnosis prediction via medical code-centric multimodal contrastive EHR modelling with hierarchical regularisation. In *Findings of the Association for Computational Linguistics: EACL*, pages 41–55, 2024.

Yikuan Li, Shishir Rao, Jose R. A. Solares, Abdelaali Hassaine, Rohan Ramakrishnan, Dexter Canoy, Yajie Zhu, Kazem Rahimi, and Gholamreza Salimi-Khorshidi. BEHRT: Transformer for electronic health records. *Scientific Reports*, 10:7155, 2020.

Ming De Ma, Xin Wang, Yu Xiao, Alessandro Cuturrufo, Vijay S. Nori, Eran Halperin, and William Wang. Memorize

and rank: Elevating large language models for clinical diagnosis prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

Chao Pang, Xiaoqian Jiang, K. S. Kalluri, Matthew Spotnitz, Rui Chen, Adler Perotte, and Karthik Natarajan. CEHR-BERT: Incorporating temporal information from structured electronic health records to improve prediction tasks. In *Proceedings of Machine Learning for Health*, volume 158, pages 239–260, 2021.

Laila Rasmy, Yang Xiang, Zhiwen 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.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

Elitza S. Theel, Ryan D. Johnson, Elizabeth Plumhoff, and Curtis A. Hanson. Use of the Optum Labs Data Warehouse to assess test ordering patterns for diagnosis of *Helicobacter pylori* infection in the United States. *Journal of Clinical Microbiology*, 53(4):1358–1360, 2015.

Du, N., Huang, Y., Dai, A. M., et al. GLaM: Efficient Scaling of Language Models with Mixture-of-Experts. In *Proceedings of ICML*, 2022.

Songtao Jiang, Tuo Zheng, Yan Zhang, Yeying Jin, Li Yuan, and Zuozhu Liu. Med-MoE: Mixture of domain-specific experts for lightweight medical vision-language models. In *Findings of the Association for Computational Linguistics: EMNLP*, pages 3843–3860, 2024.

Jiang, A. Q., Sablayrolles, A., Mensch, A., et al. Mixtral of Experts. *arXiv preprint arXiv:2401.04088*, 2024.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021.

Tang, S., Wang, Y., et al. Self-supervised Transformer for Electronic Health Record Representation Learning. *Nature Communications*, 2023.

Wornow, M., Thapa, R., et al. Foresight: Generative Pretraining for Patient Trajectories. *The Lancet Digital Health*, 2024.

Anonymous. CEHR-GPT: Foundation Models for Electronic Health Records. *arXiv preprint*, 2025.

Anonymous. EHRMamba: Scalable State Space Models for Electronic Health Records. *arXiv preprint arXiv:2405.14567*, 2024.

A. Extended Experimental Details

Structured tokenization and sequence construction. Each patient record is serialized as a chronological typed-token timeline. Static demographic attributes are prepended as tokens (for example age bucket, sex, and region), followed by day-level diagnosis and procedure bundles separated by explicit begin/end delimiters. Diagnosis tokens are standardized to a compact ICD-oriented vocabulary and procedure tokens to a canonical procedure vocabulary. This design keeps the token space structured enough for transfer while remaining compatible with autoregressive pretraining.

OLDW incident-diabetes endpoint. The diabetes endpoint is sequence-based incident diabetes prediction. Positives have at least one of 48 diabetes ICD codes in the held-out post-history bundle and no diabetes code in the prior history; negatives have no diabetes code anywhere in the full sequence. We require at least three ICD bundles of history before applying the endpoint rule. The resulting cohort is small and balanced at the patient-timeline level, so all diabetes results should be interpreted as supportive validation rather than a definitive clinical-risk benchmark.

Zero-shot MIMIC-IV harmonization. The public evaluation pipeline follows four steps. (1) Patient, admission, diagnosis, and procedure tables are joined into temporally ordered timelines. (2) Legacy diagnosis and procedure systems are mapped into the ICD-10-oriented and canonical procedure spaces used by SparseEHR. (3) The resulting sequences are formatted into the same typed-token schema used during OLDW pretraining. (4) Records containing invalid or unresolved tokens are removed before evaluation. The current export retains 515 patient-level timelines after these filters, which limits statistical precision and comparability. Importantly, the conversion pipeline is frozen before testing and does not use MIMIC labels for model selection or adaptation.

Code-mapping audit. Legacy-code harmonization introduces an uncontrolled source of benchmark variation. In a stratified ICD-9-to-ICD-10 audit against GEMs-style mappings, the conversion matched the correct three-character ICD-10 category in 79% of 100 reviewed codes and 84.6% after excluding E-codes. For diabetes-related codes, 24/40 mappings were unambiguous, while 16/40 had one-to-many candidates; the assisted selection differed from the first deterministic candidate for 10 codes, mostly at sub-code specificity rather than broad phenotype level. For procedure-code harmonization, the reviewed mappings agreed on all 19 procedures, although 36.8% required disambiguation. We therefore treat the MIMIC-IV comparison as evidence of promising zero-shot transfer under our fixed mapping protocol, not as a definitive protocol-matched benchmark.

Why we use Recall@K. Next-visit diagnosis prediction is a multi-label ranking problem with a large, sparse output space: each visit contains only a few true codes out of many possible codes in the model vocabulary. We therefore report Recall@K, which asks whether the correct code appears in the top-K predictions at each autoregressive step. Recall@K is useful for ranking evaluation, but it is not sufficient by itself for clinical risk prediction or protocol-matched state-of-the-art claims; precision-oriented metrics, calibration, and matched preprocessing should be added in future benchmark work.

B. Additional Ablation Tables

Stored versus active parameter accounting. Let L be the number of transformer layers, D the number of dense-prefix layers, E the number of router-selectable experts, and k the number of selected routed experts. We use

$$P_{\text{stored}} = P_{\text{base}} + DP_{\text{denseFFN}} + (L - D)\left(P_{\text{shared}} + EP_{\text{routed}} + P_{\text{router}}\right), \tag{5}$$

$$P_{\text{active}} = P_{\text{base}} + DP_{\text{denseFFN}} + (L - D)\left(P_{\text{shared}} + kP_{\text{routed}} + P_{\text{router}}\right), \tag{6}$$

where P_{base} contains embeddings, attention/projection matrices, normalization, and other always-executed non-expert components. For SparseEHR, L=16, D=4, E=4, and k=1.

C. Qualitative Routing Summary

Table 8 condenses the supporting routed-token analysis from the full draft. The goal of this appendix is not to claim that each expert maps to one disease family. Instead, the evidence points to *partial specialization by EHR function*: some experts emphasize common diagnosis backbone patterns, while others focus more strongly on workflow, procedure transitions, or context-specific chronic care.

Table 3. Component-level parameter accounting used in the implementation audit. This table states what is included in stored and active counts without relying on a blanket E/k compute-reduction shorthand.

Component	Stored parameter count	Active parameter count per token
Embeddings, attention/projection, norms, and routers	Counted once in stored model capacity.	Counted because these components are executed for each token.
Dense prefix FFNs	Counted for each of the first D dense layers.	Counted because all tokens pass through the dense prefix.
Always-active shared experts	Counted for each MoE layer.	Counted because every token is processed by the shared expert.
Router-selectable experts	All E routed experts are counted in stored capacity for each MoE layer.	Only the selected k routed expert(s) are counted for the token; inactive routed experts are excluded from active computation.

Table 4. Validation sensitivity of the diabetes decision threshold k_{cls}. The selected value k_{cls}=50 maximizes balanced accuracy on the held-out validation set.

k_{cls}	Precision	Recall	Accuracy	F1	Bal. Acc.
1	1.000	0.009	0.508	0.017	0.504
3	1.000	0.085	0.546	0.156	0.542
5	1.000	0.136	0.571	0.239	0.568
10	0.946	0.297	0.643	0.452	0.640
20	0.907	0.415	0.689	0.570	0.687
50	0.771	0.700	0.746	0.734	0.746
100	0.667	0.898	0.727	0.765	0.728

Table 5. Structured-EHR baselines for the OLDW incident-diabetes endpoint. XGBoost is the strongest classical baseline and is competitive with the SparseEHR thresholded code-ranking rule, so diabetes results are treated as supportive rather than as evidence of definitive superiority.

Method	F1
Logistic Regression	0.613
LightGBM	0.599
XGBoost	0.732
SparseEHR (k_{cls}=50)	0.735

Table 6. Prompted-LLM references and SparseEHR under a shared hard binary endpoint. GPT models are retained as secondary references, not primary structured-EHR baselines. SparseEHR uses k_{cls}=50, selected on validation balanced accuracy.

Method	Acc.	Prec.	Rec.	F1	Bal. Acc.
GPT-4.1	0.530	0.680	0.386	0.493	0.562
GPT-5	0.600	0.590	0.580	0.585	0.600
SparseEHR (k_{cls}=50)	0.748	0.769	0.703	0.735	0.748

Table 7. Rare-code versus common-code recall on the diabetes subset for the selected model. Rare-code prediction remains harder, but recall increases steadily with larger K, indicating useful long-tail retrieval.

K	Overall R@K	Rare-code R@K	Common-code R@K
1	0.1996	0.1027	0.2027
3	0.3096	0.2169	0.3035
5	0.3812	0.2769	0.3742
10	0.4720	0.3260	0.4890
20	0.5546	0.3761	0.5955
50	0.6903	0.4812	0.7454
100	0.7707	0.5684	0.8369

Interpretation. This pattern is important for structured-health modeling because it suggests that strong load balancing does not destroy clinically meaningful specialization. Instead, the model shares ubiquitous high-frequency tokens across

Table 8. Representative routed-token summaries for four MoE layers in the selected D=4 model (expanded version)

Layer	Summary
4	The first MoE layer after the dense warm start splits common diagnosis-sequence backbone tokens from procedure-heavy and workflow-linked context such as lab procedures, office-visit patterns, demographics, and visit markers.
6	Expert roles become sharper: one expert emphasizes cardiometabolic follow-up, another remains a general chronic-diagnosis backbone, a third leans toward musculoskeletal and skin-related chronic care, and a fourth captures mixed screening and outpatient workflow.
11	Routing shows both sharing and specialization. Common chronic-diagnosis tokens remain distributed across multiple experts, while other experts emphasize procedure-linked transitions, preventive care patterns, and lower-frequency contextual content.
15	The final MoE layer shows the most mature specialization by EHR function: a common diagnosis backbone, visit-transition patterns, chronic ambulatory care, and outpatient management/lab-office workflow.

experts while still reserving capacity for lower-frequency workflow and chronic-care contexts. That is a plausible and desirable compromise for large structured EHR corpora, where common recurring codes and rare contextual events coexist in the same patient trajectory.

D. Appendix Discussion

The appendix results reinforce the main paper's bounded claim. First, the parameter accounting distinguishes stored expert capacity from active per-token computation and explicitly includes the always-active shared expert, selected routed expert, dense FFNs, attention/projection weights, embeddings, routers, and normalization components. Second, zero-shot transfer is compatible with a fully code-based evaluation protocol, which is appealing for structured-data settings where notes or local labels may be unavailable, but the MIMIC comparison is not protocol-matched to published baselines. Third, the model's long-tail behavior is imperfect but non-trivial: rare-code recall remains below common-code recall at every cutoff, yet it grows substantially as K increases. Together, these details support conditional computation as a promising capacity/transfer trade-off for structured health data, not as a universal efficiency or state-of-the-art claim.