From Binning to Joint Embeddings: Robust Numeric Integration for EHR Transformers

Maria Elkjær Montgomery
Department of Computer Science
University of Copenhagen
maem@di.ku.dk

Mads Nielsen
Department of Computer Science
University of Copenhagen
madsn@di.ku.dk

Abstract

Transformer-based EHR models are sensitive to how numeric values are represented. Using synthetic and real data, we investigate the effects of binning numeric values, as well as the effects of inputting floating point values directly into the EHR sequence (optionally after binning) or making a joint continuous-categorical embedding. We uncover a simple scaling rule: the optimal bin count follows a power law with dataset size. Joint embeddings deliver the highest accuracy and robustness to noise, while direct injection is also effective, especially when preceded by binning. Models perform surprisingly well on complex arithmetic tasks, which illustrates that although they cannot perform exact computation, this may not be necessary in an EHR setting where measurements by nature are noisy. On two clinical prediction tasks, adding lab measurements yields small but consistent AUROC gains. These results offer practical guidance for numeric integration and a path toward multimodal EHR transformers.

1 Introduction

Transformer-based architectures are now central to predictive modelling with Electronic Health Records (EHR) [22], often outperforming classic ML methods [9, 14]. BERT-inspired models (e.g., BEHRT [10], Med-BERT [17]) showed that contextual code embeddings capture temporal dependencies. Subsequent efforts addressed temporal dynamics, long sequences, and model optimisation for clinical prediction [14, 24, 13, 21]. Decoder-style models extend these ideas to generative patient modelling [15, 8]. Still, most large-scale EHR transformers remain limited to discrete categorical inputs (diagnoses, procedures, medications), underutilising continuous numeric data such as laboratory values and vital parameters, despite these being abundant, less biased, and highly informative for clinical interpretation. Incorporating numeric values also provides a natural bridge to multimodal modelling by enabling the integration of other quantitative sources (e.g. imaging-derived measurements). Strategies for numeric integration include discretisation [19, 18, 11], dedicated numeric embeddings [10], and joint categorical-continuous representations [2]. Reported effectiveness varies, and the conditions under which each approach succeeds remain unclear, particularly across clinical contexts where numeric signals can reflect single measurements, combinations, or trends. In this work, we (1) implement and compare five approaches for embedding numeric values in transformer-based EHR models, (2) evaluate each approach under controlled settings, and (3) test real-world clinical prediction tasks.

2 Related work

Many studies have explored how to handle both continuous and discrete inputs within transformer architectures. TabTransformer [6] applies attention only to categorical features, concatenating their

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Multimodal Representation Learning for Healthcare @ EurIPS 2025.

contextualised embeddings with numeric inputs before prediction. FT-Transformer [5] and SAINT [20] extend this idea by tokenising both feature types and feeding them jointly into a transformer encoder. More recently, xVal [4] extends numeric representations by introducing a dedicated artificial token for each numeric feature, whose embedding is scaled by the feature's actual value. In the EHR domain, Labrador [2] adopts an approach conceptually similar to FT-Transformer and xVal, where categorical and continuous features each pass through dedicated embedding layers, and their outputs are summed to form joint embeddings. However, it does not incorporate other EHR modalities. Alternative methods discretise continuous inputs [19, 11, 18], either through clinically defined thresholds or quantile binning, but these may not generalise across heterogeneous numeric features. Overall, several strategies exist for encoding numeric data, but few have been tested systematically in an EHR context. We aim to evaluate how these methods perform across different numeric inputs and how such representations can serve as a foundation for multimodal patient modelling.

3 Methods

3.1 Experimental Setup

We use a BERT-style model similar to the one described in CORE-BEHRT [13] but with a ModernBERT [23] backbone instead. The model has hidden size 96, intermediate size 192, six layers, six heads, and a 1024 token context window. We evaluate different strategies for incorporating numeric values using both synthetic and real-world data. Synthetic data enables controlled variation in feature count, interactions, and temporal structure, while real-world data validate findings under clinical noise and dependencies. The real-world data is derived from the electronic health record system used across the Capital Region and Region Zealand in Denmark. It includes records from all patients who interacted with the the hospitals in these regions between 2016 and 2024, comprising a total of around 2.2 million unique individuals. To isolate the impact of value representation, we insert synthetic lab measurements and corresponding outcome labels into each patient's timeline between birth and (if applicable) death, ensuring that numeric inputs appear in realistic temporal contexts. Each label is placed 10–180 days after the last synthetic lab measurement, and all events within 48 hours prior to the label are censored to prevent leakage. The model predicts the label from the remaining EHR history.

Baseline experiments We evaluate each method on datasets of size 10,000, 100,000, and 1,000,000, split 50-40-10 into pre-training, fine-tuning, and test sets. Experiments are conducted on three class conditional Gaussian settings with fixed standard deviation 10 and varying mean separation. For label $y \in \{0, 1\}$ we sample

$$x \mid y = 0 \sim \mathcal{N}(\mu_0, 10^2), \quad x \mid y = 1 \sim \mathcal{N}(\mu_1, 10^2)$$

where $(\mu_0, \mu_1) \in \{(35, 65), (45, 55), (48, 52)\}$. Because the synthetic lab values are the sole signal for predicting y, we compute the theoretical optimal ROC AUC for each setup, see appendix A. Performance is then measured by (i) the absolute gap to this optimum, d, and (ii) the number of standard deviations from the optimum, d_σ, with standard deviation computed using Delong's test [25]. We fix the standard deviation at 10 (varying SD added no benefit over varying mean separation), and use a 50/50 class split (performance was limited by minority-class size, not imbalance). From the baseline experiments, we carry forward the best-performing and conceptually diverse methods to the arithmetic and clinical evaluations.

Arithmetic experiments To assess the limits of each method, we construct arithmetic tasks of increasing complexity, each formulated as predicting whether $f(LAB_{1:n}) > \text{median}(f(LAB_{1:n}))$. This creates a binary prediction task requiring the model to combine multiple synthetic lab values. The tasks include **counting** (frequency under the baseline Gaussian setup - difficulty via overlap set by μ_0 and μ_1), **addition** ($\sum LAB_n$), **multiplication** ($\prod LAB_n$), and **polynomial** evaluation (all monomials up to degree d, where where the total number of terms is given by $\leq d$: $T(n, d) = \binom{n+d}{d}$, where n is the number of lab measurements). For addition/multiplication we add input noise. All arithmetic tasks are scored with d and d_σ.

Clinical experiments Finally, to evaluate whether findings from synthetic tasks generalise to real-world prediction, we test the methods on clinical outcomes using the real-world data exclusively.

Specifically, we assess performance on breast cancer and lung cancer prediction tasks, examining whether the inclusion of lab test values embedded through each method improves model performance.

3.2 Data representations

We evaluate five strategies for representing continuous values in transformer-based EHR models:

- **Binning**: Continuous values are binned into categorical tokens and discretised.
- **Combined**: Categorical and continuous features are embedded separately: Both are then passed to the transformer.
- **Combined binned**: Combined approach, but continuous values are binned first.
- **Concat**: Categorical and continuous embeddings are concatenated and passed through a projection layer to form a joint feature representation.
- **FiLM**: Categorical and continuous embeddings are merged through feature-wise linear modulation (FiLM) [16], using learnable scaling and shifting parameters.

All numeric inputs are min-max normalised before embedding. A more detailed description the methods can be found in appendix B. For the **binning** method we systematically evaluated different bin counts across varying sample sizes and value distributions to identify a generalisable binning strategy based on data availability. The resulting optimal binning configuration is then reused in the **combined binned** method. Across all methods, a masked pre-training objective is applied to both categorical and continuous components, with joint optimisation of classification and regression losses for all methods except **discrete**, where only a classification loss is used.

4 Results

To derive a general rule for binning continuous values, we evaluated bin counts (3, 5, 10, 25, 50, 75, and 100) across varying sample sizes and value distributions in the baseline setup. Each experiment was repeated five times, and we generated 300 additional simulations from the resulting mean and standard deviation (appendix C). For each distribution, we computed the average d and d_σ, and identified the optimal bin count by minimising d across settings. This was then fit to a power-law of the form $k = an^b$, yielding $4.85n^{0.163}$, see figure 1. This fit was motivated by theoretical scaling laws such as Scott's rule for histogram binning [1].

Figure 2 shows baseline results on all five methods on the $(\mu_0, \mu_1) = (48, 52)$. This setup highlights where the methods differ the most and is the most comparable to real clinical prediction scenarios. From these results, we selected **FiLM**, **combined binned**, and **binning** for continued experimentation. We chose FiLM over concat since the two performed similarly, but FiLM provides additional interpretability. For the arithmetic experiments, figure 3 reports performance on the counting (frequency) task across Gaussian class separations defined by μ_0 and μ_1. Next, figure 5 shows the performance of the methods on the multiplication task, where increasing task complexity eventually pushes all methods to failure. Results for the addition tasks followed a similar pattern. Finally, figure 4 shows model performance across polynomial degrees (d), with shaded areas indicating standard deviation. Results are averaged over runs with $n = 2, 3,$ and 4 lab measurements. Clinical results shown in appendix D show small improvements from adding lab tests. Combined binned performs best with AUROC rises from 0.660 to 0.712 for breast cancer and from 0.785 to 0.787 for lung cancer.

5 Discussion and conclusion

Our results show that representation choices for numeric data shape performance and generalisability in transformer-based EHR models. We identify a simple rule for discretisation: the optimal number of bins grows with dataset size according to a power-law. Binning notably improved the combined approach in baseline experiments, especially under limited data, suggesting it stabilises learning when numeric signals are sparse or noisy. Joint-embedding methods (Concat, FiLM) also performed strongly. This is noteworthy given results such as CEHR-BERT [14], where injecting additional tokens (for time) into the primary sequence outperformed adding temporal features in an additional

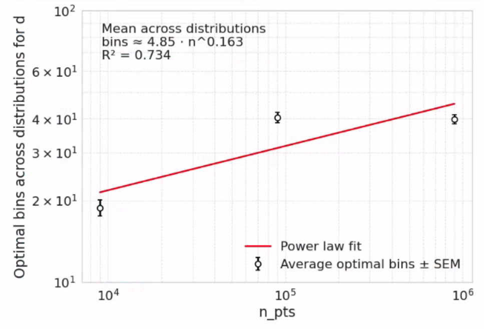


Figure 1: Fitted power law for optimal bin count across dataset sizes.

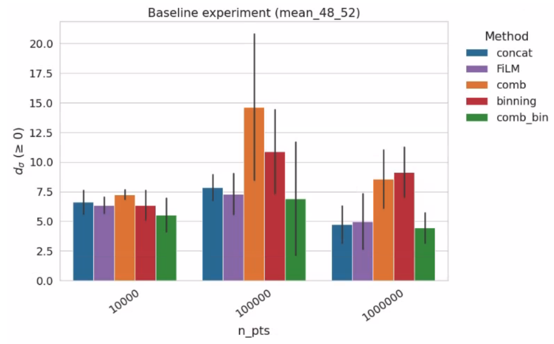


Figure 2: Baseline performance (d_σ) for all five methods with mean separation $(\mu_0, \mu_1) = (48, 52)$.

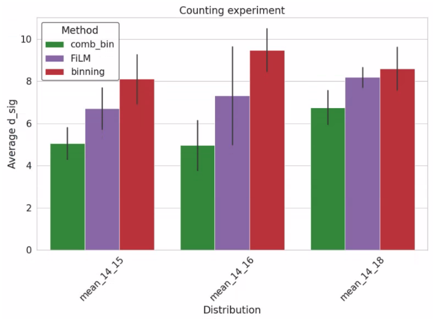


Figure 3: Performance on the counting task.

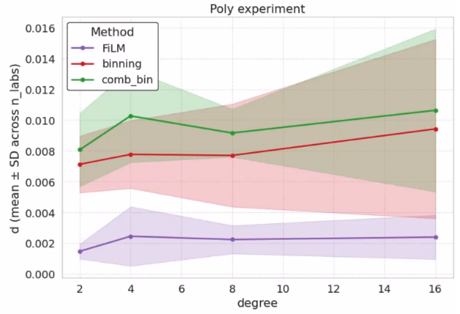


Figure 4: Performance on the polynomial task.

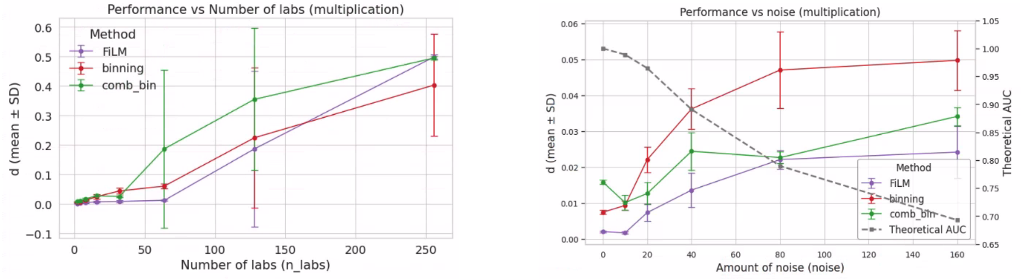


Figure 5: Model performance on the multiplication task. (a) Effect of increasing the number of lab values. (b) Effect of increasing input noise.

embedding layer. We expected a similar advantage for numeric inputs, yet Concat/FiLM consistently showed impressive performance. Still, methods that avoid extra embedding layers remain attractive in practice as they scale cleanly as modalities grow, and are easier to adapt to decoder-style or multimodal architectures. The models handled surprisingly complex arithmetic despite prior reports that transformers struggle with numerical reasoning [7, 12, 3]. In the multiplication experiments, performance degraded gradually rather than collapsing at the heuristic limit of 2^L with $L = 6$ being the number of transformer layers. This likely reflects that much of the literature targets exact arithmetic, whereas our aim tolerate "good enough" accuracy for EHR use cases. Accordingly, the choice of representation should be guided by the required precision and the target architecture. Clinical results show modest but consistent gains from adding lab measurements, with combined binned strongest.

In conclusion, the results offer practical guidance for integrating numeric values in transformer-based EHR models and a simple binning rule that scales with data size, with implications for multimodal extensions. Limitations include use of a single encoder-style backbone (ModernBERT), few runs per setting, and a limited clinical evaluation (two tasks within one regional hospital network). Future work should extend to alternative architectures (decoder, encoder–decoder, multi-modal), broader clinical testing, and larger repeated experiments to reduce variance and assess generalisability.

References

[1] Histogram, howpublished = `https://www.graphmatik.io/docs/charts/histogram`, note = Accessed: 2025-10-09.

[2] David R Bellamy, Bhawesh Kumar, Cindy Wang, and Andrew Beam. Labrador: Exploring the limits of masked language modeling for laboratory data. *arXiv preprint arXiv:2312.11502*, 2023.

[3] Shaoxiong Duan, Yining Shi, and Wei Xu. From interpolation to extrapolation: Complete length generalization for arithmetic transformers, 2024.

[4] S Golkar, M Pettee, M Eickenberg, A Bietti, M Cranmer, G Krawezik, F Lanusse, M Mc-Cabe, R Ohana, L Parker, et al. xval: A continuous numerical tokenization for scientific language models (2024). *arXiv preprint arXiv:2310.02989*.

[5] Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. *Advances in neural information processing systems*, 34:18932–18943, 2021.

[6] Xin Huang, Ashish Khetan, Milan Cvitkovic, and Zohar Karnin. Tabtransformer: Tabular data modeling using contextual embeddings. *arXiv preprint arXiv:2012.06678*, 2020.

[7] Samy Jelassi, Stéphane d'Ascoli, Carles Domingo-Enrich, Yuhuai Wu, Yuanzhi Li, and François Charton. Length generalization in arithmetic transformers, 2023.

[8] Zeljko Kraljevic, Dan Bean, Anthony Shek, Rebecca Bendayan, Harry Hemingway, Joshua Au Yeung, Alexander Deng, Alfred Balston, Jack Ross, Esther Idowu, et al. Foresight—a generative pretrained transformer for modelling of patient timelines using electronic health records: a retrospective modelling study. *The Lancet Digital Health*, 6(4):e281–e290, 2024.

[9] Manuel Lentzen, Thomas Linden, Sai Veeranki, Sumit Madan, Diether Kramer, Werner Leodolter, and Holger Fröhlich. A transformer-based model trained on large scale claims data for prediction of severe covid-19 disease progression. *IEEE journal of biomedical and health informatics*, 27(9):4548–4558, 2023.

[10] Yikuan Li, Shishir Rao, José Roberto Ayala Solares, Abdelaali Hassaine, Rema Ramakrishnan, Dexter Canoy, Yajie Zhu, Kazem Rahimi, and Gholamreza Salimi-Khorshidi. Behrt: transformer for electronic health records. *Scientific reports*, 10(1):7155, 2020.

[11] Ndèye Maguette Mbaye, Michael Danziger, Aullène Toussaint, Elise Dumas, Julien Guerin, Anne-Sophie Hamy-Petit, Fabien Reyal, Michal Rosen-Zvi, and Chloé-Agathe Azencott. Multimodal behrt: Transformers for multimodal electronic health records to predict breast cancer prognosis. *medRxiv*, pages 2024–09, 2024.

[12] Rodrigo Nogueira, Zhiying Jiang, and Jimmy Lin. Investigating the limitations of transformers with simple arithmetic tasks. *arXiv preprint arXiv:2102.13019*, 2021.

[13] Mikkel Odgaard, Kiril Vadimovic Klein, Sanne Møller Thysen, Espen Jimenez-Solem, Martin Sillesen, and Mads Nielsen. Core-behrt: A carefully optimized and rigorously evaluated behrt. *arXiv preprint arXiv:2404.15201*, 2024.

[14] Chao Pang, Xinzhuo Jiang, Krishna S Kalluri, Matthew Spotnitz, RuiJun Chen, Adler Perotte, and Karthik Natarajan. Cehr-bert: Incorporating temporal information from structured ehr data to improve prediction tasks. In *Machine Learning for Health*, pages 239–260. PMLR, 2021.

[15] Chao Pang, Jiheum Park, Xinzhuo Jiang, Nishanth Parameshwar Pavinkurve, Krishna S Kalluri, Shalmali Joshi, NoÃ Elhadad, Karthik Natarajan, et al. Cehr-gpt: A scalable multi-task foundation model for electronic health records. *arXiv preprint arXiv:2509.03643*, 2025.

[16] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[17] 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.

[18] P Renc et al. Zero shot health trajectory prediction using transformer. npj digit. *Med*, 7:1–10, 2024.

[19] Lorenzo A Rossi, Chad Shawber, Janet Munu, and Finly Zachariah. Evaluation of embeddings of laboratory test codes for patients at a cancer center. *arXiv preprint arXiv:1907.09600*, 2019.

[20] Gowthami Somepalli, Micah Goldblum, Avi Schwarzschild, C Bayan Bruss, and Tom Goldstein. Saint: Improved neural networks for tabular data via row attention and contrastive pre-training. *arXiv preprint arXiv:2106.01342*, 2021.

[21] Ethan Steinberg, Jason Fries, Yizhe Xu, and Nigam Shah. Motor: A time-to-event foundation model for structured medical records. *arXiv preprint arXiv:2301.03150*, 2023.

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[23] Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference, 2024.

[24] 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. *arXiv preprint arXiv:2412.16178*, 2024.

[25] yandexdataschool. roc_comparison. `https://github.com/yandexdataschool/roc_comparison`, 2021.

A Computation of theoretical AUC

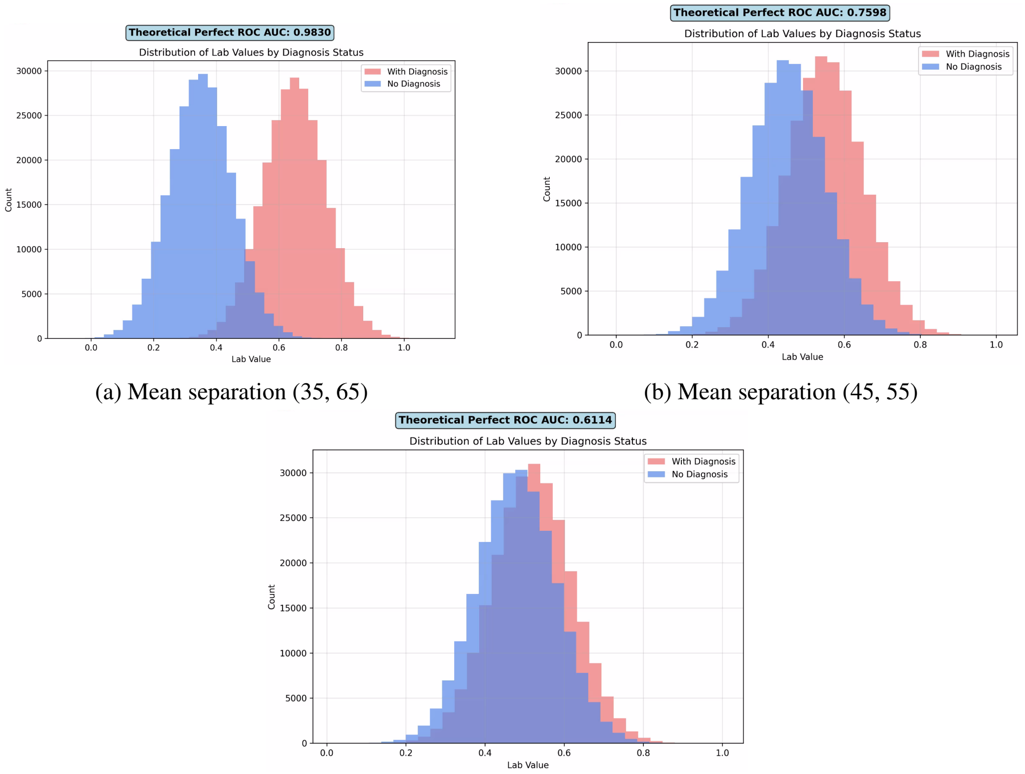
Figure 6 shows the theoretical ROC AUC values for the baseline setting with $n = 100,000$. In this setting, the synthetic lab measurement is the only feature that carries information about the assigned label, which allows us to compute the theoretical maximum achievable ROC AUC. These theoretical values therefore represent the upper bound on performance for any model in this setup.

B Details on the numeric integration methods

Figure 7 summarises the four numeric-integration schemes visualised in this paper: *Discrete*, *Combined*, *Concat*, and *FiLM*. All methods follow the CORE-BEHRT architecture [13]: the full EHR sequence is used as model input, the model is first trained with masked language modelling (MLM), and is then fine-tuned with a prediction head for the downstream task.

B.1 Discretisation

In the **discrete** method, continuous inputs are converted into categorical tokens through binning. Specifically, each value is first normalised, multiplied by the number of bins, and then mapped to a token of the form `VAL_X`, where X denotes the corresponding bin index. These tokens are then treated as standard inputs during pre-training, which follows a masked language modelling (MLM) objective where masked tokens are predicted using a cross-entropy loss.



(a) Mean separation (35, 65)

(b) Mean separation (45, 55)

(c) Mean separation (48, 52)

Figure 6: The theoretical ROC AUC for the baseline Gaussian experiments at three levels of class-mean separation.

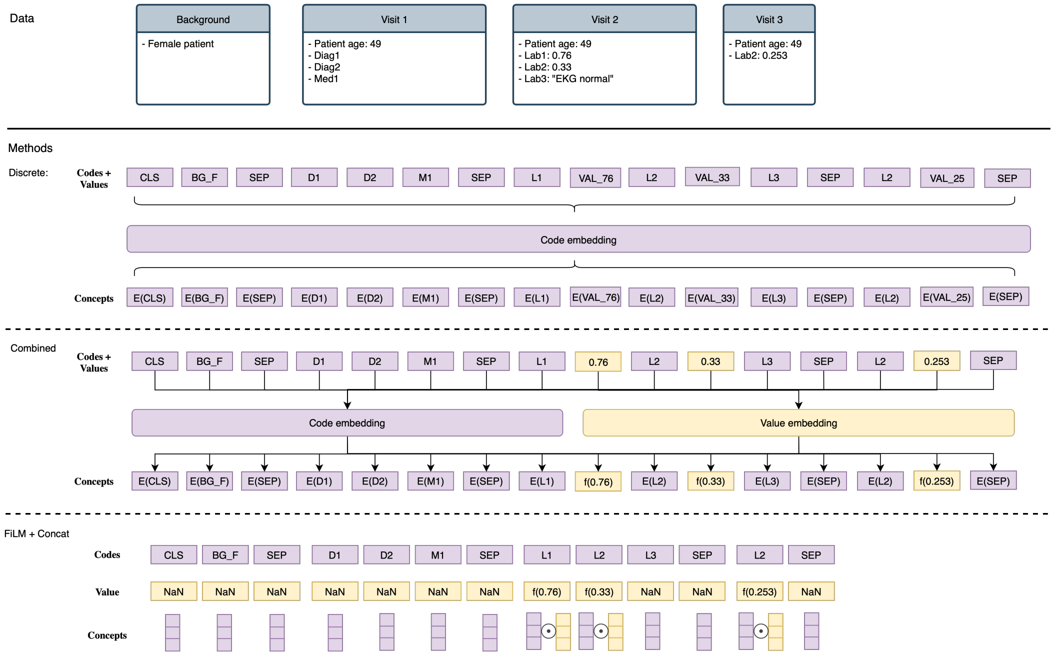


Figure 7: Overview of four numeric-integration schemes for EHR transformers: (a) **Discrete** (bin values into tokens), (b) **Combined** (parallel categorical/value streams), (c) **Concat** (concatenate then project), and (d) **FiLM** (feature-wise linear modulation). Numeric inputs are min–max normalised.

B.2 Combined

In the **combined** approach, categorical and continuous features are embedded separately. Categorical features are embedded using a discrete embedding layer into a vector of dimension d, while continuous features are projected into the same dimension d using a linear layer. During pre-training, both categorical and continuous values are masked. The model is trained with a dual prediction objective: categorical values are predicted through a classifier head using a cross-entropy loss, while continuous values are predicted through a regression head using a mean-square-error loss (MSE). The two objectives are optimised jointly using a combined loss function. The loss starts with equal weighting for categorical and continuous components, but the weight for the continuous loss is made learnable, allowing the model to adapt its balance during training.

B.3 Concat and FiLM

In the **concat** and **FiLM** methods, a separate value layer is introduced, where numeric values are aligned with their corresponding categorical features. If a categorical feature has no associated continuous value, a NaN placeholder is assigned in the value layer. Categorical features are embedded using a discrete embedding layer, while non-NaN numeric values are projected through a linear layer. For features with both categorical and continuous components, these representations are combined into a joint embedding; for features with only categorical values, the categorical embedding is used directly.

In the **concat** method, the joint embedding is formed by concatenating the categorical and continuous embeddings, followed by a projection layer.

In the **FiLM** method, the joint embedding is computed using Feature-wise Linear Modulation:

$$\mathbf{e}_{\text{joint}} = \gamma(\mathbf{e}_{cat}) \cdot \mathbf{e}_{cont} + \beta(\mathbf{e}_{cat}),$$

where $\mathbf{e}_{cat}$ is the categorical embedding, $\mathbf{e}_{cont}$ is the continuous embedding, and γ, β are learnable functions applied to $\mathbf{e}_{cat}$.

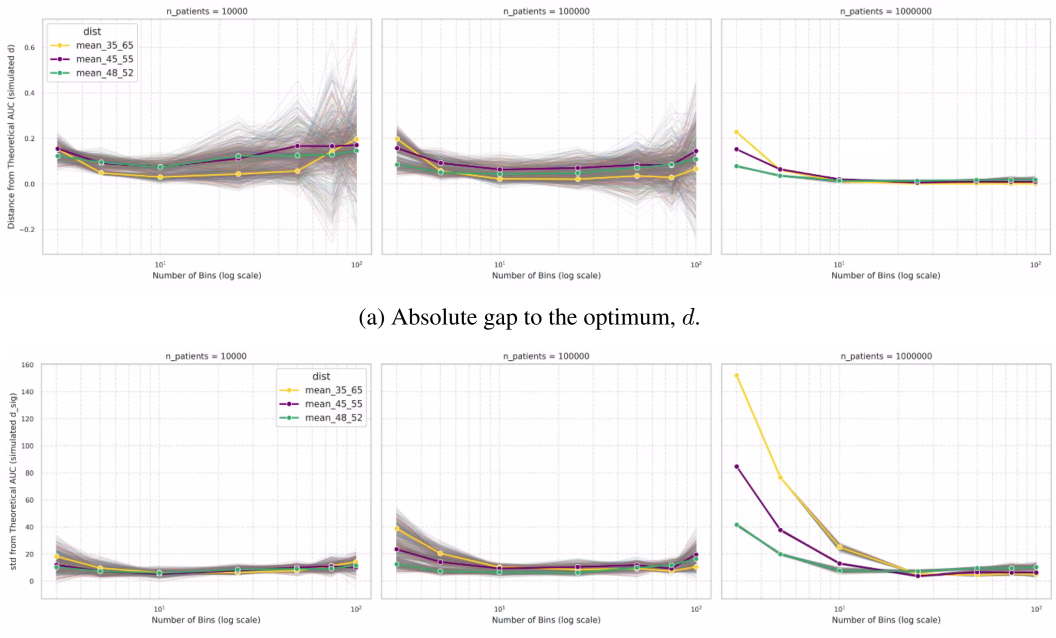
During pre-training, for both methods, a continuous value is masked whenever its associated categorical value is masked. The model is then trained to jointly predict both components: categorical values are predicted through a classification head using cross-entropy loss, while continuous values are predicted through a regression head a MSE-loss. The two losses are combined into a single objective. Initially, categorical and continuous losses are weighted equally, but the weight for the continuous loss is made learnable.

C Simulation experiments for optimal binning

Figure 8 reports baseline results for the *Discrete* method across bin counts {3, 5, 10, 25, 50, 75, 100}, dataset sizes, and Gaussian mean separations. Each configuration is run five times to estimate mean and standard deviation. We then generate 300 simulations from these estimates. Panel (a) shows the absolute gap to the theoretical optimum (d); panel (b) shows the standardised gap (d_σ).

D Clinical experiments

Figure 9 reports clinical results as the change in AUROC relative to a model without lab inputs on the two tasks breast cancer prediction and lung cancer prediction. We also include a control that adds only lab test names (no numeric values).



(a) Absolute gap to the optimum, d.

(b) Standardised gap, d_σ.

Figure 8: Discrete-method binning across dataset sizes and Gaussian separations. Points show run means (5 repeats). Grey curves show 300 simulated draws from these estimates.

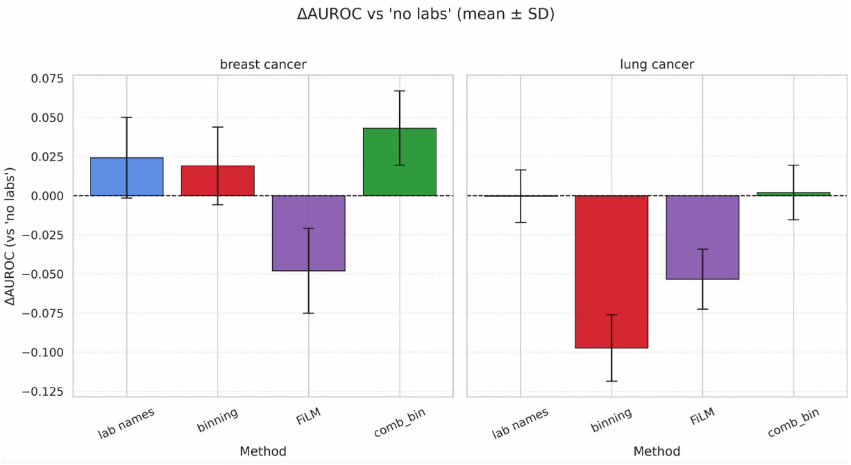


Figure 9: Clinical evaluation: change in AUROC relative to a "no labs" baseline. Positive values indicate an increase in AUROC; negative values indicate a decrease.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract claims that we evaluate five methods for integrating numeric values into transformer-based EHR models, identify when each succeeds or fails, and test whether these findings generalise to clinical prediction tasks. The paper supports this by comparing methods across controlled data distributions, quantifying performance with d and d_σ and testing limits via binning choices (including an empirically fitted power-law rule), stress tests with noise, and increasing term counts. We also show that the methods handle complex arithmetic insufficient for exact computation, but suitable for EHR settings where robust, approximate accuracy is often adequate, thereby informing multimodal use cases. Finally, we evaluate two clinical prediction tasks (breast and lung cancer), where adding lab measurement yeilded a small performance gain. These claims are however limited to encoder-style transformers and the datasets studied. As we did not evaluate other architectures or broader modalities beyond the proposed numeric-integration pathways, we cannot state anything on this.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The conclusion explicitly lists limitations: reliance on a single encoder-style backbone (ModernBERT), a limited number of runs per setting, and a narrow clinical evaluation.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA] .

Justification: There are no theoretical proofs.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The results are partially reproducible. We release code for the model and for generating the synthetic datasets, enabling end-to-end replication of the controlled experiments in the supplementary material (`https://github.com/Montgomeryyyy/BONSAI_values`). The real-world EHR data cannot be shared for privacy reasons, but we provide the full training/evaluation pipeline and configuration details so authorised holders of comparable EHR data can reproduce those experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code can be found at `https://github.com/Montgomeryyyy/BONSAI_values`, and all the configuration files corresponding to the experiments can also be found there.

   The health records utilised in this study were acquired from hospitals in the Region of Zealand and the Capital Region of Denmark, which was approved by the Danish Patients Safety Board (Styrelsen for Patientssikkerhed, approval #31-1521-182) and the Danish Capital Region Data Safety Board (Videncenter for data-anmeldelser, approval #P-2020-180). Anyone wanting access to the data will be required to meet research credentialing requirements as outlined on the web site: `https://www.regionh.dk/til-fagfolk/Forskning-og-innovation/Hvilke-tilladelser-kraever-dit-projekt-/Sider/Forskningsprojekter-baseret-p%C3%A5-registerdata-og-journaldata.aspx`. Requests are normally processed within 3-6 months.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes config files are available in the supplementary material.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Yes error bars in form of standard deviations (SD) or standard deviations of the mean (SEM) are shown where relevant.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: No. Due to this being run with low-priority nodes on a private Microsoft Azure cloud, we cannot easily measure the resources/time/etc the experiments need.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Analyses are retrospective and non-interventional. Only de-identified hospital EHR were used under data-use agreements. No patient data is shared.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: No not explicitly, the paper just focuses on reporting the performance of the different methods on mostly synthetic data tasks.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [No]

    Justification: Due to the nature of that data, the pre-trained models and data will not be publicly available.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The code will be uploaded in supplementary material anonymised. This includes documentation to the extent that is possible.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.