# Comparative Analysis of Foundation Models for Hospital Integration

Stanford CS224N Custom Project

**Suhana Bedi, Miguel Fuentes**
Department of Biomedical Data Science,
Department of Computer Science
Stanford University
suhana@stanford.edu, migufuen@stanford.edu

## Abstract

Annually, hospitals produce an estimated 50 petabytes of electronic health record (EHR) data, much of which remains unexplored. Foundation models present a significant opportunity to leverage this abundant, unlabeled data to enhance patient care. Research has shown that transformer-based foundation models are capable of developing robust patient representations, thereby increasing the accuracy of clinical prediction models. However, the complex nature of patient data, which typically consists of long sequences of timestamped structured and unstructured text, presents a substantial challenge. This complexity often surpasses the processing capabilities of conventional transformer models like GPT, which struggle with extended context lengths due to quadratic scaling issues. This study addresses this limitation by comparing the traditional GPT model—a decoder-only transformer—with Mamba, a selective state space model that scales linearly with context length. We evaluate these models on validation perplexity and their effectiveness across 15 few-shot clinical prediction tasks. Our analysis aims to highlight the advantages of foundation models in terms of scaling efficiency and their ability to adapt to specific clinical tasks.

## 1 Key Information to include

- Mentor: Sonia Hangjie Chu
- Sharing project: Shared with external research project

## 2 Introduction

Electronic health record (EHR) systems are extensively used in the United States (Henry et al. (2016), and the vast volume of data they produce presents a valuable opportunity for machine learning-based predictive modeling to improve clinical decision support. In particular, clinical prediction models have been deployed to forecast in-hospital mortality (Li et al. (2021), diagnoses (Zhang et al. (2020), and length of stay (Cai et al. (2016); Harerimana et al. (2021).

EHRs include structured data and unstructured data. While unstructured data mostly consists of clinical notes, structured data includes important information like vital signs (e.g., heart rate, respiration rate, temperature, and blood pressure), diagnosis, procedure and medicine-related information. However, the EHR data of a patient is high dimensional and sparse. Each patient can have multiple hospital visits which are irregularly spread over the years such that each visit has some or all of the clinical data described above. This makes it difficult to deploy off-the-shelf machine learning models like logistic regression or decision trees that expect a vector of features with fixed length Steinberg et al. (2020).

Recent work has been done train deep neural networks directly on raw patient EHRs for prediction tasks like hospital readmission or mortality. This approach shows limited improvement in a clinical setting because deep learning models typically require large datasets, and EHR datasets are often constrained by the limited number of patients with specific clinical outcomes Huang et al. (2021).

Transfer learning is an effective strategy in Natural Language Processing (NLP) and computer vision to address issues around small datasets Hosna et al. (2022). It typically involves pretraining a model on a large dataset and then fine-tuning it on a smaller one based on the task at hand. The choice of the pre-training task is critical, with language modeling being a popular option that involves learning a generative sequence model for text. Representation learning, a subset of transfer learning, focuses on creating fixed-length representations that can be reused for various tasks. This is particularly beneficial for EHR data, where patient representations can be generated through language modeling and then reused for a myriad of tasks since the number of patients available for specific outcomes is usually limited compared to the overall patient population in a healthcare system.

Recent work Niu et al. (2024); Pang et al. (2024) has leveraged Transformer-based models, such as BERT and GPT, to generate fixed-length vector representations of patients by applying language modeling techniques and fine-tuning these representations to predict clinical outcomes. Despite their high accuracy, deploying these models in real clinical settings is challenging due to specific architectural constraints inherent to Transformers. The self-attention mechanism at the core of Transformer models allows each token to attend to every other token, capturing complex dependencies within the data. However, this mechanism leads to quadratic scaling of computational and memory requirements with respect to the context length, resulting in a computational and memory complexity that scales as $O(n^2)$ Vaswani et al. (2023). This becomes prohibitive when processing patient medical records that span tens of thousands of tokens to capture an entire medical history, as the computational and financial costs escalate, and increased latency becomes an issue in real-time clinical environments.

For instance, the task of predicting heart failure readmission within 30 days requires analyzing comprehensive, longitudinal EHR data, including chronic conditions, medication adherence, acute events, and lifestyle factors recorded over several years. The limited context window of standard Transformer models necessitates truncating patient records, potentially omitting critical information and leading to a drop in predictive accuracy. The quadratic scaling issue of self-attention mechanisms makes it impractical to capture the entirety of a patient's medical history, affecting the model's performance on specific clinical tasks.

To address these challenges, we explore attention-free models, namely Mamba and HyenaGu and Dao (2023); Poli et al. (2023). These models do not rely on the self-attention mechanism and therefore do not suffer from the quadratic scaling issue. Mamba, a type of state space model, uses a combination of linear operations and non-linear functions to update the state representation which leads to linear scaling ($O(n)$) with sequence length. Hyena, on the other hand, leverages a subquadratic-time complexity approach, specifically $O(NL \log_2 L)$ where $N$ is the sequence length and $L$ is the filter size in its convolutional layers. This model employs implicit long convolutions and element-wise gating to handle longer context lengths more efficiently. As a result, Hyena achieves faster computation times and lower memory requirements compared to traditional attention-based models.

## 3 Related Work

### 3.1 Deep Neural Network Based Clinical Prediction Models Using EHR Data

Recent advancements in clinical prediction models using EHR data have concentrated on training deep neural networks in an end-to-end fashion to predict specific outcomes. These models have been developed for various outcomes, including all-cause mortality Avati et al. (2017), heart failure Choi et al. (2016b), COPD, unplanned readmissions Cheng et al. (2016), and future hospital admissions. Typically, these studies introduce innovative neural network architectures and report performance improvements over baseline models.

### 3.2 Language model based representation learning

Representation learning is widely used in computer vision and NLP to address the challenge of limited training data. In the context of EHR data, representation learning techniques often draw from

natural language processing due to structural similarities. Just as a text document is a sequence of words with learned representations for individual words or entire sentences, a patient's longitudinal EHR can be viewed as a "document" comprising a sequence of diagnoses, procedures, medications, and laboratory results. This approach focuses on the structured data within EHRs rather than the textual content of clinical notes. A language model is a probabilistic framework for sequences of words, typically implemented as a neural network with millions of parameters. These models capture the language generation process by predicting one word at a time, thereby modeling the likelihood of word sequences. Previous work Choi et al. (2016a); Steinberg et al. (2021)has shown that language model based representations are significantly better than a wide array of alternative representations for training clinical prediction models across a range of training set sizes.

### 3.3 Attention-free models

State space models (SSMs) Gu et al. (2022) are a class of models that represent a system using a set of hidden states and observations over time. In the context of EHR data, SSMs can be used to capture the temporal dynamics and latent structures within patient records. These models are particularly useful for handling irregularly spaced data and can provide a more nuanced understanding of patient trajectories compared to traditional models. One of the key advantages of SSMs is their linear scaling with context length, which makes them computationally efficient for long sequences of data. This linear scaling can allow SSMs to process extensive patient histories without the prohibitive computational and memory requirements faced by models like transformers, which scale quadratically. Another replacement for the attention operator is Hyena, a long convolution and element-wise multiplicative gating operator. Hyena operators are able to significantly shrink the quality gap with attention at scale, reaching similar perplexity and downstream performance with a smaller computational budget Poli et al. (2023). Therefore, the use of attention-free models for generating patient representations remains unexplored yet promising due to predicted computational efficiency gains.

## 4 Approach

First, we obtain electronic health record (EHR) data from the Stanford Medicine Research Data Repository (STARR). Typically, each patient $P$ is represented as a sequence of medical events $[E_1, E_2, \ldots, E_N]$, where $N$ is the total number of medical events in the patient's record. For each event $E_i$, there is a set of associated features, which include diagnoses, procedures, medication orders, and laboratory test orders (encoded using ICD10, CPT or HCPCS, RXCUI, and LOINC codes). Following the approach of Choi et al. Hur et al. (2023), we map each medical code to its textual description. These features are then tokenized into a sequence of sub-words. An event encoder $f$ subsequently converts the sequence into an embedding $m_i$. With embeddings for each event $E_i$, we feed these embeddings into our model of choice to generate patient representations. Finally, we use the generated patient representations as input to a logistic regression model to predict clinical outcomes. In this study, we used PyTorch Lightning to manage and optimize the training of our models, which were sourced from the Hugging Face Model Hub and initialized from scratch. PyTorch Lightning allowed us to define our models in a structured and modular manner, encapsulating the model architecture, training steps, and validation routines within a **pl.LightningModule**. The training configuration, including optimizer setup and learning rate scheduling, was simplified through PyTorch Lightning's high-level interfaces. The Trainer class handled the training process, managing aspects like logging, checkpointing, and early stopping, thereby automating and improving efficiency. This framework significantly reduced boilerplate code, enhanced readability and organization, and facilitated scaling to multi-GPU and distributed training environments, making the training process more effective and manageable.

Next, we establish GPT Brown et al. (2020) as our baseline model, in accordance with previous studies that demonstrate its efficacy in language modeling and clinical prediction tasks Wornow et al. (2023); Steinberg et al. (2023). This sets GPT as a proven benchmark within the field. Moreover, to compare the performance of different architectures, we include the Mamba Gu and Dao (2023) and Hyena models Nguyen et al. (2023).

Upon selecting the models for pretraining and evaluation, we perform learning rate sweeps over 10 epochs for Mamba, Hyena, and GPT. This process allows us to systematically explore the effect

of different learning rates on model performance, ensuring that we identify an optimal learning rate for each model. We then report the training and validation perplexities for the best-performing learning rates for all models in Figure 1. Additionally, we include AUROC graphs for the downstream tasks derived from EHRSHOT Wornow et al. (2023), a benchmark designed for evaluating model performance on a variety of clinical prediction tasks. We selected this benchmark because of the diversity of tasks it evaluates, including operational outcomes, laboratory results, diagnosis assignment and x-ray findings on a combination of binary, multiclass and multilabel classification.

# 5 Experiments

## 5.1 Data

The experiments utilized de-identified electronic health record (EHR) data from Stanford Hospital and Lucile Packard Children's Hospital Datta et al. (2020). This dataset includes 3.6 million patient records collected between 1990 and 2018. We preprocessed that data using FEMR (cite femr), a python package for manipulating longitudinal EHR data for machine learning, with a focus on supporting the creation of foundation models. We did some exploratory data analysis to understand the demographic distribution (gender, race, and ethnicity) of our patient population. More info regarding the patient demographic distribution can be found in Appendix A. After preprocessing and preliminary data analysis, we performed a 70, 15 and 15% split for training, validation and test sets.

## 5.2 Evaluation

We use three evaluation metrics to comprehensively assess the models' performance. First, we evaluate language modeling performance using validation perplexity. Perplexity measures the degree of uncertainty a language model has when generating a new token, providing insight into how well the model predicts the next word in a sequence. Second, we measure the time taken to pretrain a model for 10 epochs with a context length of 1024 tokens. This metric reflects the computational efficiency of different model architectures. Lastly, we assess the robustness of the patient representations generated by the models using the AUROC (Area Under the Receiver Operating Characteristic curve) on clinical prediction tasks derived from EHRSHOT. The AUROC metric helps us understand how effectively the models can distinguish between different clinical outcomes, reflecting their utility in real-world healthcare scenarios.

## 5.3 Experimental details

We trained our models for 10 epochs using the base configurations for GPT2-Base, Mamba-Tiny and Hyena-Medium. In all cases, we used a set of three learning rates: 1e-4, 1e-5, 1e-6 and a context length of 1024 tokens. Table 1 contains a summary of the model configurations used for training.

Table 1: Experimental Setup and Model Configurations

| Model | Parameter Count | Model Configuration and Hyperparameters |
|---|---|---|
| **GPT2-Base** | 214 M | n_embed: 768, n_head: 12, n_layer: 12 |
| **Mamba-Tiny** | 218 M | d_model: 768, n_layer: 24, num_hidden_layers: 24, state_size: 16 |
| **Hyena-Medium** | 92.2 M | d_model: 256, n_layer: 8 |

The **n_embed** parameter refers to the dimension of the token embeddings, indicating how each token is represented as a vector. The **n_head** parameter specifies the number of attention heads in the multi-head attention mechanism, allowing the model to focus on different parts of the input sequence simultaneously. The **n_layer** parameter denotes the number of layers in the model, defining its depth. The **d_model** parameter represents the dimension of the model's hidden states, which are the internal representations processed by each layer. The **num_hidden_layers** parameter indicates the number of hidden layers within each model layer, contributing to the complexity of the transformations applied. Finally, the **state_size** parameter refers to the size of the state vector in state space models or recurrent architectures, determining the dimensionality of the maintained state. The optimization and hyperparameter configurations are present in table 2

Table 2: Optimization and Hyperparameter Configurations

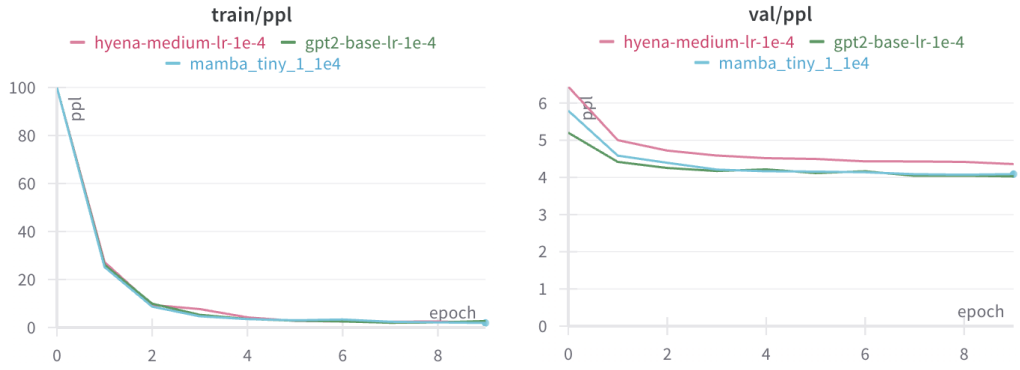
| Hyperparameter | Description |
|---|---|
| **Optimizer** | AdamW optimizer used to enhance training stability and performance. |
| **Gradient Clipping** | Gradient clipping set at 1.0 to prevent exploding gradients, stabilizing the training process. |
| **Weight Decay** | Weight decay of 0.1 applied to regularize the model and reduce overfitting. |
| **Dropout** | No dropout used, as preliminary experiments indicated better performance without it. |
| **Learning Rate** | Linear learning rate starting at 1e-4, gradually decaying to 1e-6 using a cosine decay schedule. |
| **Learning Rate Schedule** | Cosine decay schedule allows the model to converge smoothly and prevents overfitting in the later stages of training. |

### 5.4 Results

After performing learning rate sweeps, we found 1e-4 to be the best learning rate for all three models. Appendix B contains graphs for the learning rate sweeps. Figure 1 shows training and validation perplexity for GPT, Mamba and Hyena models. We observe that Mamba and GPT show a lower validation perplexity (4.089 and 4.027 respectively) while Hyena shows a slightly higher perplexity of 4.359.

In terms of time taken to train each of the models, we observe that Hyena takes the least amount of time to train for 10 epochs, while Mamba takes 200 hours to train while GPT takes 300 hours to train.

Lastly, we evaluated the models on four widely studied classification tasks in the medical domain: intensive care unit (ICU) admission, long lengths of stay (LOS), 30-day readmission, and celiac disease diagnosis. The performance metrics, shown in the AUROC graphs in Figure 2 , demonstrate that Mamba outperforms GPT2-Base. These tasks serve as critical benchmarks, encompassing a diverse range of clinical scenarios from acute critical care to chronic disease management, thereby providing a comprehensive assessment of the models' generalization capabilities across varied medical conditions. For a complete evaluation of the models' performance on all 15 downstream tasks, please refer to Appendix C.




(a) Train ppl

(b) Val ppl

Figure 1: Best GPT2-Base (lr=2e-4) and Mamba-Tiny (lr=1e-4). X axis is the number of training steps and y axis is perplexity (ppl). Validation ppl was computed twice per epoch during the training phase.

## 6 Analysis

There is a surge in the deployment of language models within the medical field. However, healthcare providers face significant financial challenges due to the immense costs associated with these technologies Hart et al. (2023). One option is to wait for EHR providers, such as Epic, to develop new functionalities, but this sector is known for its slow pace, making internal implementations

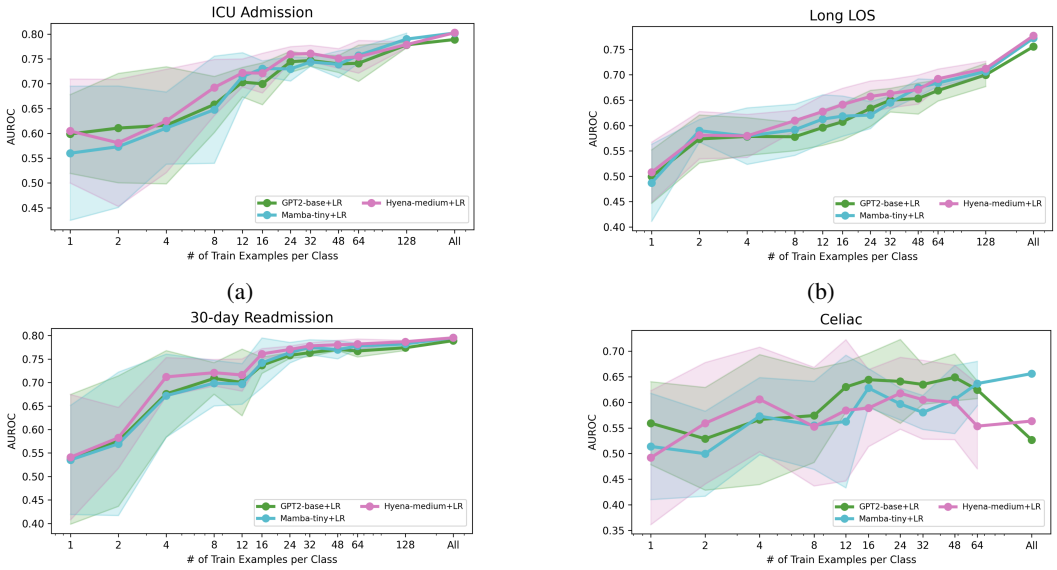



Figure 2: Mamba and Hyena perform better than GPT2 on the most common medical tasks.

more appealing. As shown in Figure 3, by the time Hyena finished its training, Mamba was on the third epoch, and GPT was still on the second one. Considering this alongside model performance, Hyena emerges as the most compelling option for decision-makers. It trains significantly faster ( 6x faster than GPT in our experiments) and demonstrates competitive performance on the EHRSHOT benchmark.

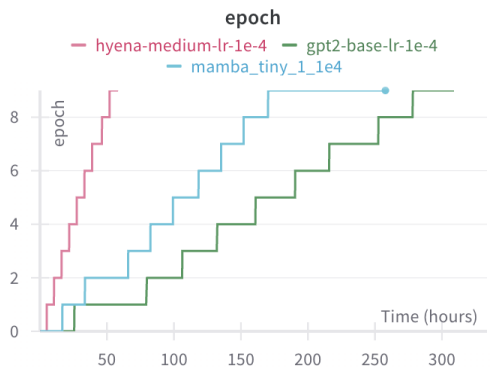



Figure 3: Hyena trains faster than both Mamba and GPT.

Here, we also notice that Mamba demonstrated superior performance on the celiac disease prediction task. This could be because of its state space architecture which is particularly effective at capturing temporal dependencies and patterns in sequential medical data. This allows it to better model the subtle and progressive nature of celiac disease, compared to GPT and Hyena. Secondly, Mamba's feature handling capabilities enable it to encode the complex relationships between various medical events, such as diagnoses, procedures, and lab results, more effectively. This nuanced encoding is vital for detecting the specific patterns and correlations indicative of celiac disease, leading to more accurate predictions.

## 7 Conclusion

In this study, we have demonstrated the potential of foundation models in leveraging the vast amounts of electronic health record (EHR) data to enhance patient care through improved clinical prediction models. By comparing the traditional GPT model with Mamba and Hyena, we aimed to address the limitations posed by the quadratic scaling of self-attention mechanisms in handling long patient records.

Our results indicate that Mamba, with its linear scaling state space model, and Hyena, with its subquadratic-time complexity approach, offer significant advantages over the conventional GPT model. Specifically, Mamba and Hyena not only achieve comparable or better validation perplexities but also exhibit superior computational efficiency. Hyena, in particular, demonstrated faster training times, making it a compelling choice for real-time clinical applications where computational resources and latency are critical considerations.

Furthermore, the performance of Mamba and Hyena on clinical prediction tasks derived from the EHRSHOT dataset underscores their robustness in generating effective patient representations. The AUROC metrics for tasks such as ICU admission, long lengths of stay, 30-day readmission, and celiac disease diagnosis confirm that these models can generalize well across varied clinical scenarios.

In conclusion, our study highlights the promise of attention-free models like Mamba and Hyena in overcoming the challenges associated with processing extensive patient histories. These models not only reduce the computational burden but also maintain high predictive accuracy, making them suitable for deployment in healthcare settings. Future work will focus on further optimizing these models and exploring their application across a broader range of clinical tasks to fully realize the potential of EHR data in improving patient outcomes.

## 8 Ethics Statement

The application of foundation models to electronic health record (EHR) data presents notable ethical issues and societal risks. Firstly, the privacy and confidentiality of patient data are of utmost concern. EHRs contain sensitive personal health information, and any breaches or unauthorized access could result in significant harm to patients. This project, which involves handling vast amounts of EHR data, raises the risk of potential data leaks or misuse. Secondly, the risk of bias and fairness in machine learning models is a critical issue. The models could inadvertently perpetuate existing biases present in historical healthcare data, leading to unequal treatment and outcomes for certain patient groups, especially marginalized communities.

To mitigate these risks, several strategies can be implemented. For privacy and confidentiality, strict data anonymization protocols should be enforced, alongside robust data security measures to ensure compliance with regulations such as HIPAA and GDPR. This includes encrypting data at rest and in transit, as well as implementing access controls to limit data exposure. For addressing bias and fairness, it is essential to employ bias mitigation techniques during model development. This can involve using diverse and representative datasets, applying fairness constraints, and continuously auditing the model's performance across different demographic groups. Additionally, involving stakeholders from diverse backgrounds in the development process can help identify and rectify biases early on. These measures will help ensure that the deployment of foundation models in healthcare is done responsibly, equitably, and securely.

## References


Anand Avati, Ken Jung, Stephanie Harman, Laura Downing, Andrew Ng, and Nigam H. Shah. 2017. Improving palliative care with deep learning. In *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 311–316. IEEE.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Xiongcai Cai, Oscar Perez-Concha, Enrico Coiera, Fernando Martin-Sanchez, Richard Day, David Roffe, and Blanca Gallego. 2016. Real-time prediction of mortality, readmission, and length of stay using electronic health record data. *Journal of the American Medical Informatics Association*, 23(3):553–561.

Yao Cheng, Fei Wang, Ping Zhang, and Jianying Hu. 2016. Risk prediction with electronic health records: A deep learning approach. In *Proceedings of the 2016 SIAM International Conference on Data Mining*, pages 432–440. SIAM.

Edward Choi, Mohammad Taha Bahadori, Andy Schuetz, Walter F. Stewart, and Jimeng Sun. 2016a. Doctor ai: Predicting clinical events via recurrent neural networks.

Edward Choi, Andy Schuetz, Walter F. Stewart, and Jimeng Sun. 2016b. Using recurrent neural network models for early detection of heart failure onset. *Journal of the American Medical Informatics Association*, 23(2):361.

Somalee Datta, Jose Posada, Garrick Olson, Wencheng Li, Ciaran O'Reilly, Deepa Balraj, Joseph Mesterhazy, Joseph Pallas, Priyamvada Desai, and Nigam Shah. 2020. A new paradigm for accelerating clinical data science at stanford medicine.

Albert Gu and Tri Dao. 2023. Mamba: Linear-time sequence modeling with selective state spaces.

Albert Gu, Karan Goel, and Christopher Ré. 2022. Efficiently modeling long sequences with structured state spaces.

Gaspard Harerimana, Jong Wook Kim, and Beakcheol Jang. 2021. A deep attention model to forecast the length of stay and the in-hospital mortality right on admission from icd codes and demographic data. *Journal of Biomedical Informatics*, 118:103778.

S. N. Hart, N. G. Hoffman, P. Gershkovich, C. Christenson, D. S. McClintock, L. J. Miller, R. Jackups, V. Azimi, N. Spies, and V. Brodsky. 2023. Organizational preparedness for the use of large language models in pathology informatics. *J Pathol Inform*, 14:100338.

JaWanna Henry, Yuriy Pylypchuk, Talisha Searcy, Vaishali Patel, et al. 2016. Adoption of electronic health record systems among us non-federal acute care hospitals: 2008–2015. *ONC data brief*, 35(35):2008–15.

Asmaul Hosna, Ethel Merry, Jigmey Gyalmo, Zulfikar Alom, Zeyar Aung, and Mohammad Abdul Azim. 2022. Transfer learning: a friendly introduction. *Journal of Big Data*, 9(1):102.

Yinan Huang, Ashna Talwar, Satabdi Chatterjee, and Rajender R. Aparasu. 2021. Application of machine learning in predicting hospital readmissions: a scoping review of the literature. *BMC Medical Research Methodology*, 21(1):96.

Kyunghoon Hur, Jungwoo Oh, Junu Kim, Jiyoun Kim, Min Jae Lee, Eunbyeol Cho, Seong-Eun Moon, Younghak Kim, and Edward Choi. 2023. Unihpf: Universal healthcare predictive framework with zero domain knowledge. *arXiv preprint arXiv:2306.12345*.

Fuhai Li, Hui Xin, Jidong Zhang, Mingqiang Fu, Jingmin Zhou, and Zhexun Lian. 2021. Prediction model of in-hospital mortality in intensive care unit patients with heart failure: machine learning-based, retrospective analysis of the mimic-iii database. *BMJ open*, 11(7):e044779.

Eric Nguyen, Michael Poli, Marjan Faizi, Armin Thomas, Callum Birch-Sykes, Michael Wornow, Aman Patel, Clayton Rabideau, Stefano Massaroli, Yoshua Bengio, Stefano Ermon, Stephen A. Baccus, and Chris Ré. 2023. Hyenadna: Long-range genomic sequence modeling at single nucleotide resolution.

H. Niu, O. A. Omitaomu, M. A. Langston, M. Olama, O. Ozmen, H. B. Klasky, A. Laurio, M. Ward, and J. Nebeker. 2024. EHR-BERT: A BERT-based model for effective anomaly detection in electronic health records. *J Biomed Inform*, 150:104605.

Chao Pang, Xinzhuo Jiang, Nishanth Parameshwar Pavinkurve, Krishna S. Kalluri, Elise L. Minto, Jason Patterson, Linying Zhang, George Hripcsak, Gamze Gürsoy, Noémie Elhadad, and Karthik Natarajan. 2024. Cehr-gpt: Generating electronic health records with chronological patient timelines.

Michael Poli, Stefano Massaroli, Eric Nguyen, Daniel Y. Fu, Tri Dao, Stephen Baccus, Yoshua Bengio, Stefano Ermon, and Christopher Ré. 2023. Hyena hierarchy: Towards larger convolutional language models. *submitted draft*. Version: submitted draft, Last Compiled: April 21, 2023.

Ethan Steinberg, Jason Fries, Yizhe Xu, and Nigam Shah. 2023. Motor: A time-to-event foundation model for structured medical records.

Ethan Steinberg, Ken Jung, Jason A. Fries, Conor K. Corbin, Stephen R. Pfohl, and Nigam H. Shah. 2020. Language models are an effective representation learning technique for electronic health record data. *arXiv preprint arXiv:2001.05295*.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need.

Michael Wornow, Rahul Thapa, Ethan Steinberg, Jason A. Fries, and Nigam H. Shah. 2023. Ehrshot: An ehr benchmark for few-shot evaluation of foundation models.

Guo-Qiang Zhang, J Feng, and A Zeng. 2020. Amia annual symposium proceedings. American Medical Informatics Association,.

# Appendix A Patient Demographics

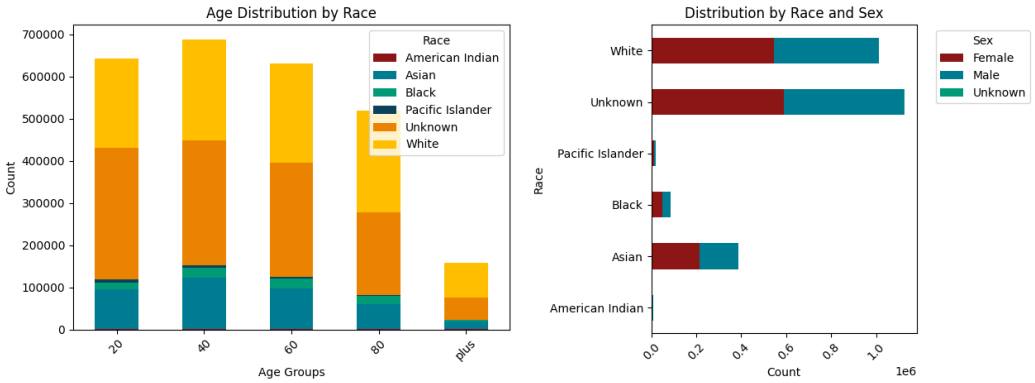
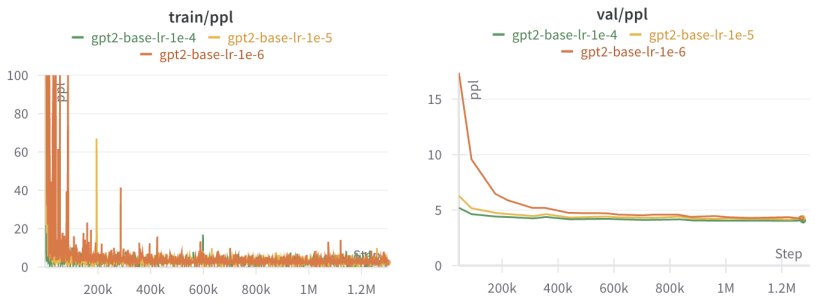


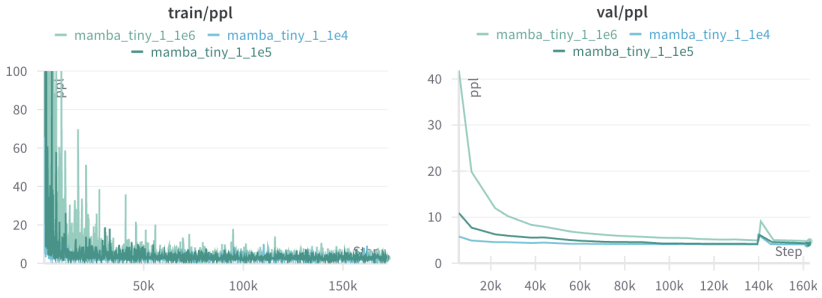

Figure 4: Patients from 20 to 60 years old represent more than 50% of the patient population. Female patients have a higher representation in the dataset.

## Appendix B Learning Rate Sweeps



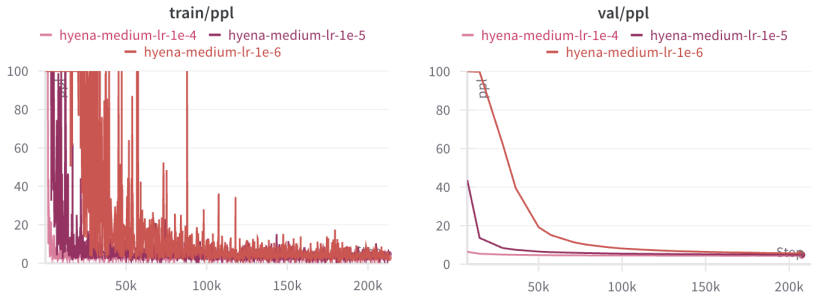

(a) Train ppl (b) Val ppl

Figure 5: GPT2-Base learning rate sweep (1e-6, 1e-5, 1e-4). All runs converge to similar perplexity values under 4.5. The best learning rate is 1e-4.




(a) Train ppl (b) Val ppl

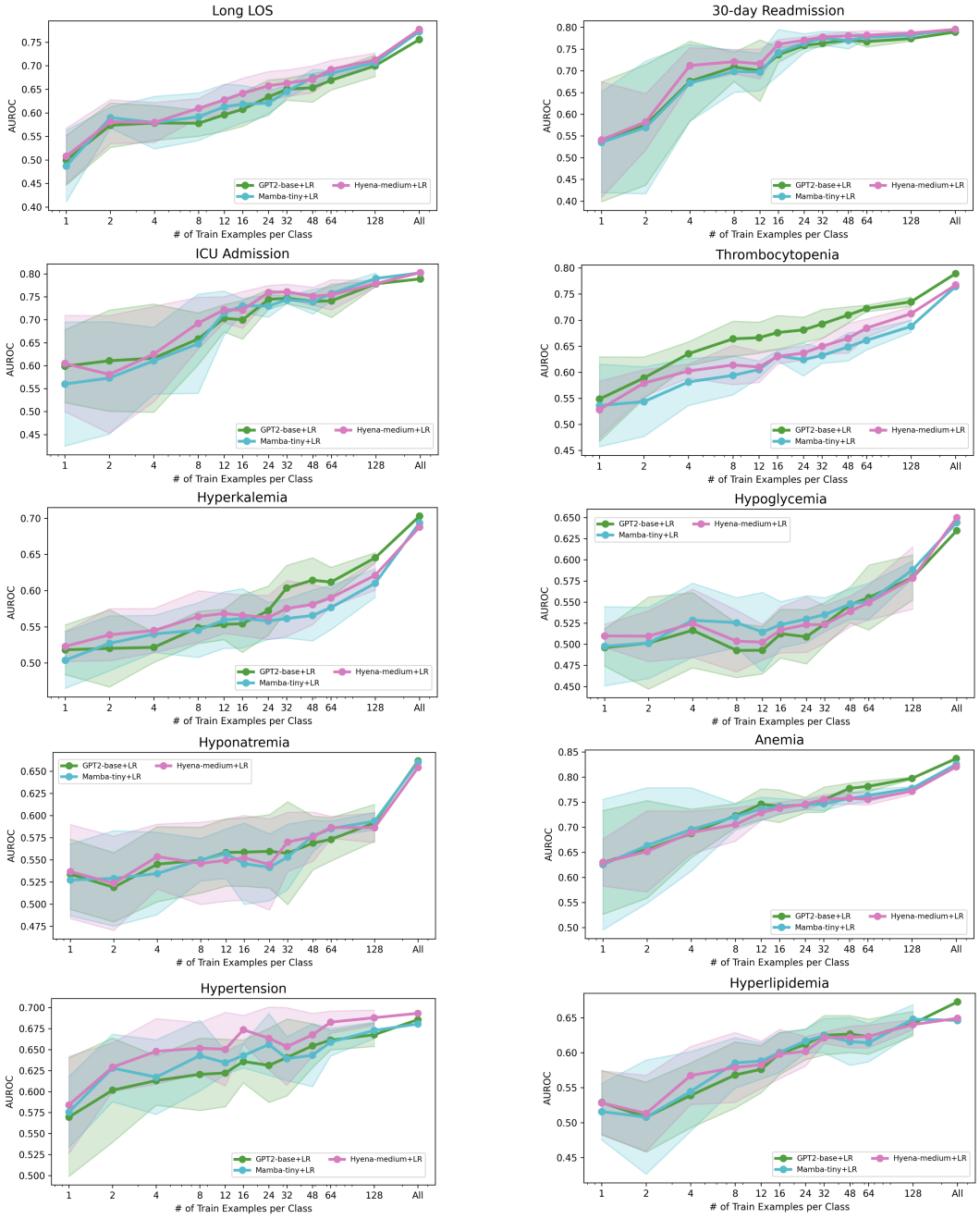
Figure 6: Mamba-Tiny learning rate sweep (1e-6, 1e-5, 1e-4). All runs converge to perplexity values close to 5. The best learning rate is 1e-4.




(a) Train ppl (b) Val ppl

Figure 7: Hyena-medium learning rate sweep (1e-6, 1e-5, 1e-4). All runs converge to perplexity values close to 5. The best learning rate is 1e-4.

# Appendix C AUROC Curves for EHRSHOT Tasks



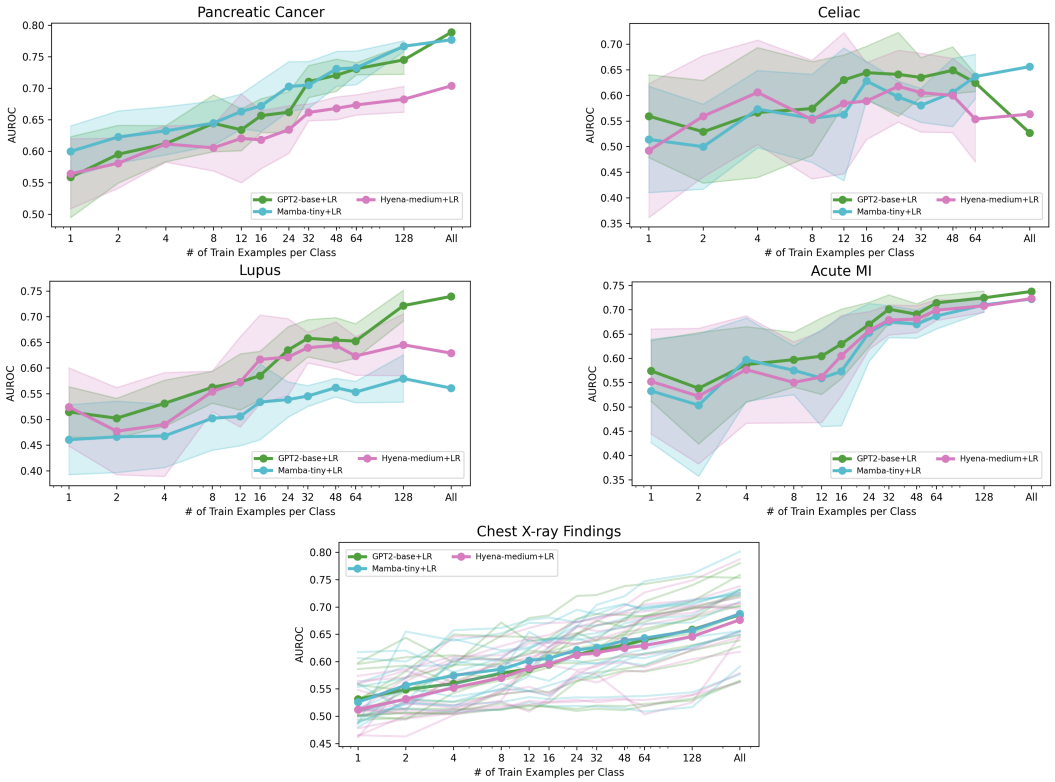



Figure 9: Performance evaluation of GPT-base, Mamba-tiny and Hyena-medium on all 15 EHRSHOT tasks.