Generating Synthetic Electronic Health Records in OMOP using GPT

Chao Pang[1], Xinzhuo Jiang[1], Nishanth Parameshwar Pavinkurve[1], Krishna S. Kalluri[1], Elise L. Minto[1], Jason Patterson[1], Karthik Natarajan[1]

1. Department of Biomedical Informatics, Columbia University

Background

Synthetic Electronic Health Record (EHR) data is crucial for advancing healthcare applications and machine learning models, particularly for researchers without direct access to healthcare systems. Although existing methods, like rule-based approaches and generative adversarial networks (GANs), generate synthetic data that resembles real-world EHR data, these methods often use a tabular format, disregarding temporal dependencies in patient histories and limiting data replication. Recently, there has been a growing interest in leveraging GPT for EHR data, considering a patient's medical history can be viewed as a document. This enables applications like disease progression analysis, population estimation, counterfactual reasoning, and synthetic data generation. In this work, we focus on synthetic data generation and demonstrate the capability of training a GPT model using a particular patient representation derived from CEHR-BERT, enabling us to generate patient sequences that can be seamlessly converted to the OMOP data format.

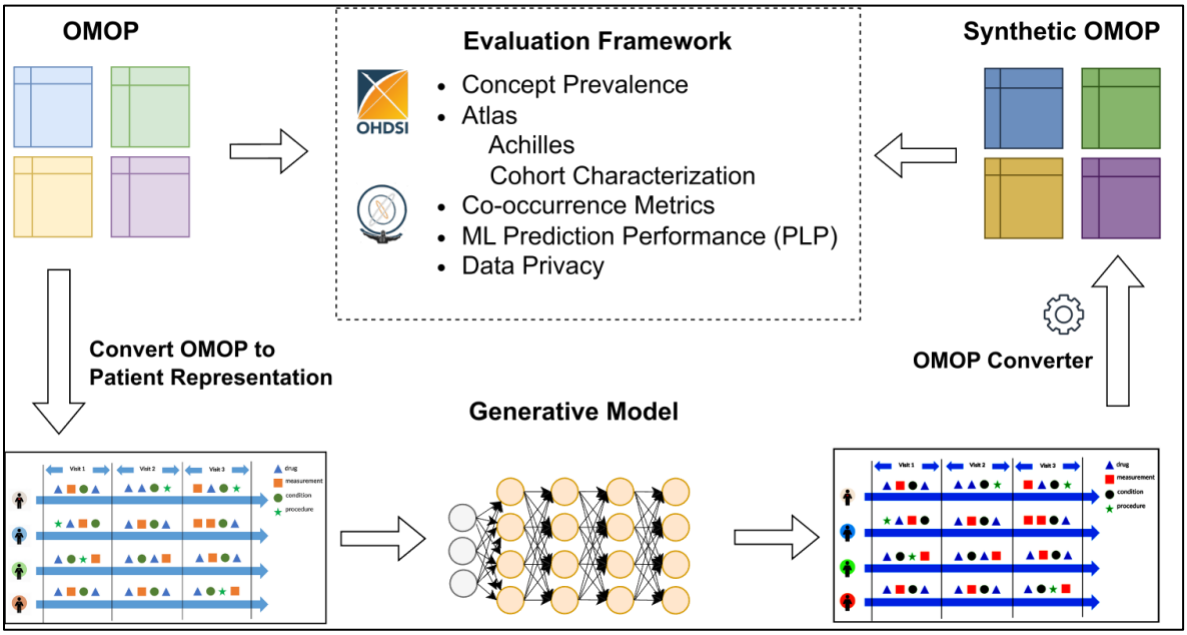

***Figure 1 Overall architecture**. The patient representation preserves demographics, visit types and temporal intervals between visits.*

Methods

In Figure 1, we present the architecture for generating synthetic data, assuming the source data is in OMOP format. To retain the temporal dependencies, we opted to work directly with time-

series patient sequences instead of converting them to a tabular format using the BOW representation. Firstly, we transformed the OMOP data into patient sequences using a particular patient representation. Secondly, we trained a generative model to learn the distribution of the patient sequences, enabling the generation of new synthetic patient sequences.
Finally, an OMOP converter was utilized to convert the synthetic patient sequences into the OMOP format. Furthermore, an evaluation procedure was developed to assess the similarity between the synthetic OMOP and the source OMOP data.

Patient Representation

The patient representation (**Figure 2**) includes demographic information, patient history, and temporal dependencies [reference]. It begins with a demographic prompt containing EHR start year, age, gender, and race. The sequence comprises visit blocks separated by artificial time tokens (ATT) representing different time intervals. Each visit block includes a visit type token to indicate the visit type.

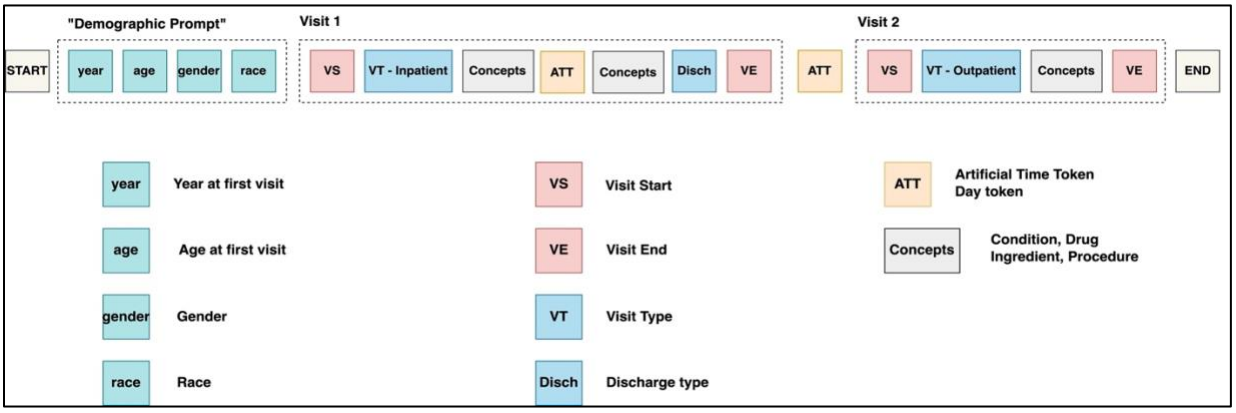

Figure 2 Patient representation*. The patient representation preserves demographics, visit types and temporal intervals between visits.*

GPT Model

We created a GPT model consisting of 6 standard transformer decoders. The input layer of the model utilized concept embedding and trainable positional embedding. When generating a patient history, we randomly sampled OMOP person records to generate demographic prompts, which served as the input to the GPT model. Using these prompts, the entire patient history was generated autoregressively by sampling tokens from the predictive distribution at the final layer.

OMOP Converter

The patient sequence was reverse engineered to convert it back to the OMOP format, shown in **Figure 3**. The start-year prompt determined the EHR history's beginning, using January 1st as the default. Demographic data was stored in the person table, while concepts were transformed into condition, drug, and procedure tables. Timestamps were calculated based on the start year and the number of days represented by each ATT token. Visits, including their visit type, were inserted into the visit table with corresponding dates. Randomness was introduced to the dates by sampling a numeric day from a uniform distribution for week, month, and long-term tokens.

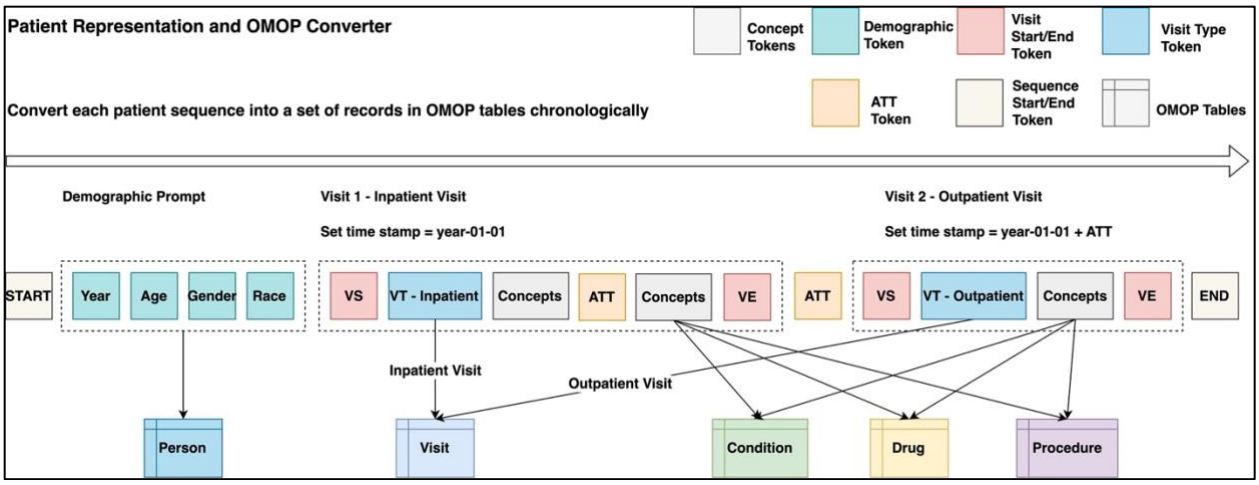

***Figure 3 OMOP Converter**. The OMOP Converter converts the patient sequences back to the OMOP format*

Evaluation procedures

We conceived an evaluation framework consisting of three levels of analyses comparing the synthetic OMOP with the original data. At the first level, the evaluation is performed to examine the concept distributions for the entire population, subpopulation (female), and specific cohorts. On the second level, we compare the co-occurrence of the concepts between the source and synthetic datasets, aiming to investigate whether the synthetic captures the correlations between concepts. The third level aims to evaluate the model's ability to reproduce machine learning prediction tasks and assess its performance. By employing this evaluation framework, the similarity between the synthetic OMOP data and the source data could be quantified.

Preliminary Results

The source patient sequences were generated from the OMOP converted from the EHR data from Columbia University Irving Medical Center-New York Presbyterian Hospital, which includes 3 million unique patients' medical histories including condition, medication, and procedure. The concept_id=0 was removed from all domains except for the visit type when constructing the patient history. We used a standard GPT model with 16 layers of transformer decoders and trained it for 2 epochs on 2 Nvidia 1080 TI GPUs with a context window of 512 tokens and a learning rate of 5e-5. We employed different sampling strategies to generate synthetic data including top k=100, top k=200, top k=300, top p=95% and top=100%. One million patient sequences were generated for each sampling strategy and converted back to synthetic OMOP instances.

Level 1 concept distribution comparison

Figure 4 shows the concept distributions between the source and the synthetic data (generated using top p=95%) across different populations and domains. Overall, the source and synthetic distributions seem to be in an alignment. In the high frequency regions, concepts tend to land on the diagonal line, indicating a good match. On the other hand, the concepts tend to spread out more in the low-frequency region. Additionally, there is an interesting cluster of condition concepts (first column) in the female population (second row) to the left side of the figure, it

turned out these were male specific conditions (pancreatic cancer) that were not supposed to be generated for the female population. The reason for this is that we have a few female patients with male specific conditions in the source data, GPT model seems to have amplified such cases. (though the percentage is very low in the synthetic data)

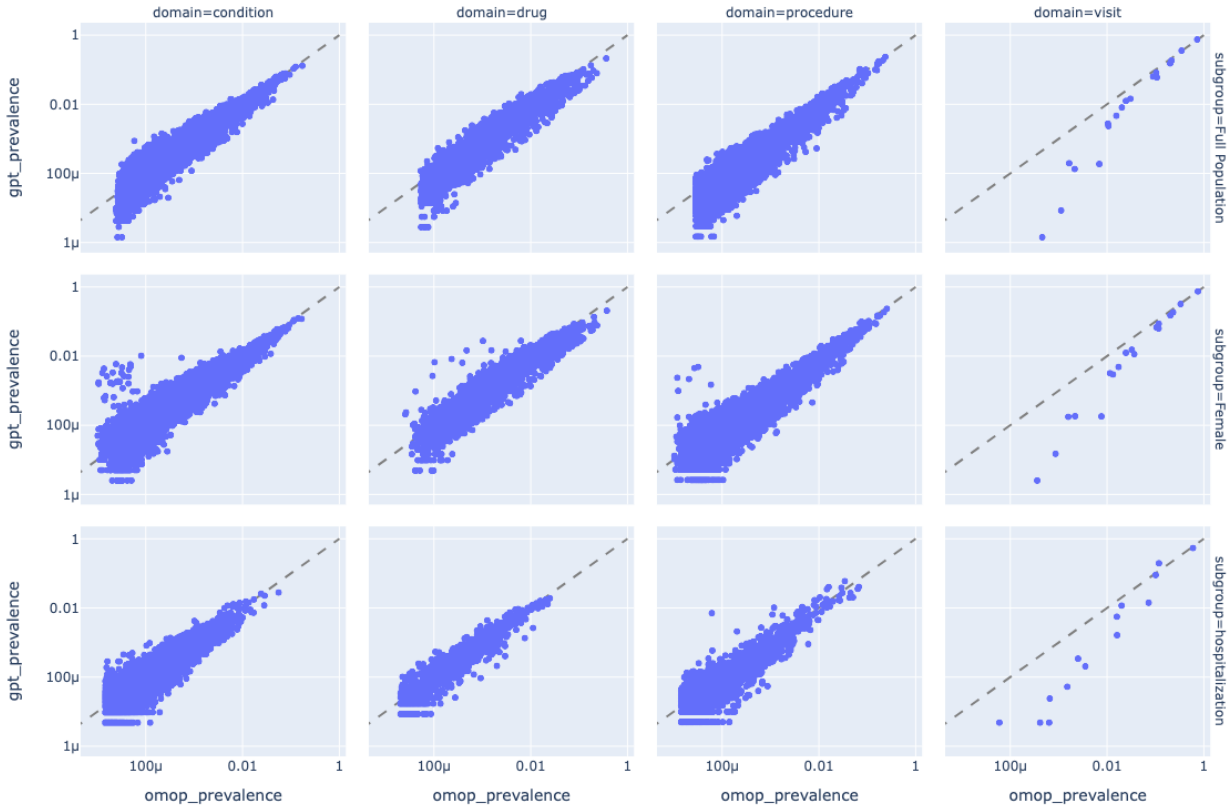

Figure 4 concept distribution comparisons between the source and synthetic data using top p=95% strategy. *The comparisons are stratified by domain (condition, drug, procedure, and visit) and population (full population, female population and hospitalization cohort) at the log scale, where x represents the source concept prevalence and y represents the synthetic concept prevalence.*

Level 2 co-occurrence comparison

We calculated the life-time co-occurrence for both source and synthetic data, where each patient was only allowed to contribute to the same concept pair once. To obtain a probability distribution, the co-occurrence matrix was normalized by its total number of records. We calculated the KL-divergence between the source and synthetic matrices to estimate the similarity between them. The procedure was applied to all synthetic datasets generated using different sampling strategies. In addition, we included two baselines, a lower and an upper bound, to better interpret the result. The lower bound was obtained by performing the same KL-divergence procedure on two random samples drawn from the source data (each consists of 10% of the source data). For upper bound, we assumed independence between any pair of concepts in the source data, and generated a hypothetical co-occurrence matrix, which was then used to calculate the KL divergence upper bound against the source co-occurrence. **Figure 5** shows the KL-divergence values for different sampling strategies, and clearly, the choice of sampling strategies affects the underlying

distribution of the synthetic data, it seems that synthetic data produced by top p=95% and top k=300 have the most similar co-occurrence matrix to the real one.

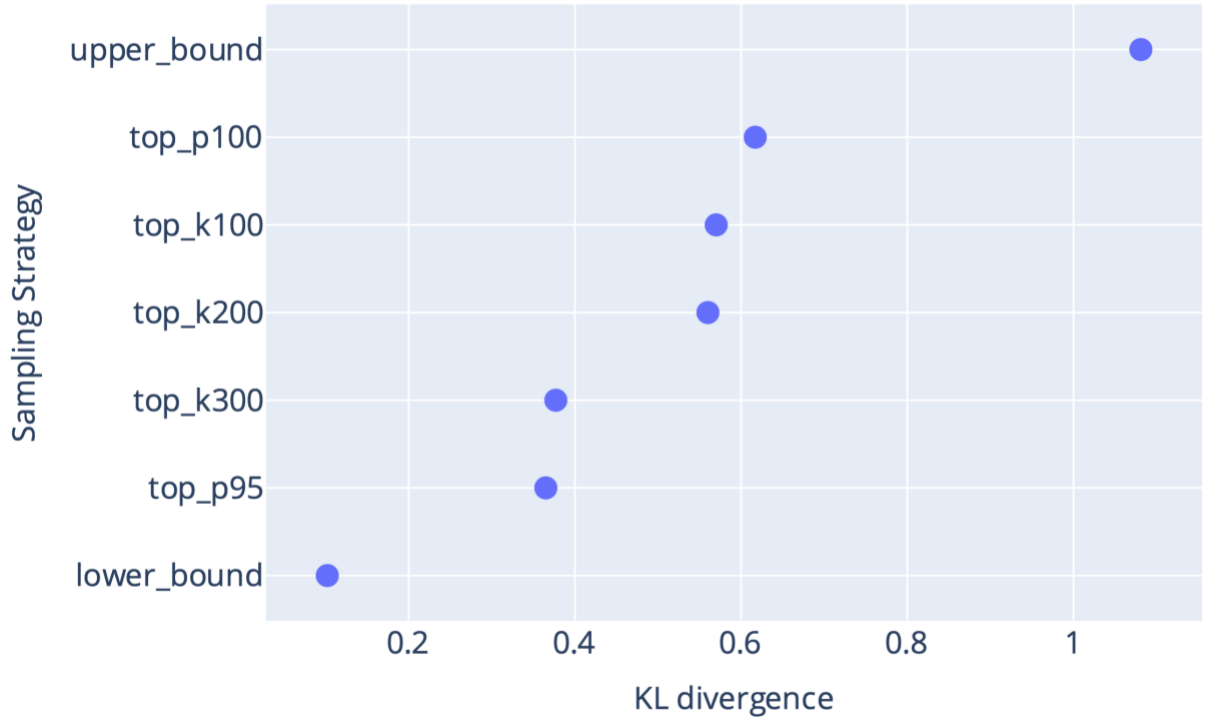

Figure 5 KL-divergence calculated between the source and synthetic co-occurrence matrices. *Two baselines are included to get a lower bound and upper bound. The lower bound was obtained by performing the same KL-divergence procedure on two random samples drawn from the source data. The upper bound was obtained by comparing the source co-occurrence with a hypothetical matrix, where all concepts were assumed to be independent of each other.*

Level 3 Logistic Regression model performance

Cohort	Cohort Definition
HF readmission	**HF patients who have a 30-day all-cause readmission. Observation window: 360 days, Prediction windows 30 days**
Hospitalization	**2-year risk of hospitalization starting from the 3rd year since the initial entry into the EHR system Observation window: 540 days, hold-off window: 180 days, Prediction windows 720 days**
COPD readmission	**COPD patients who have a 30-day all-cause readmission. Observation window: 360 days, Prediction windows 30 days**
Afib ischemic stroke	**Afib patients with 1 year risk since the initial diagnosis of afib ischemic stroke Observation window: 720 days, Prediction windows 360 day**
CAD CABG	**Patients initially diagnosed with Coronary Arterial Disease (CAD) without any prior stent graft that will receive the Coronary artery bypass surgery (CABG) treatment Observation window: 720 days, Prediction windows 360 day**

Table 1 Cohorts included for estimating the model performance

We included 5 cohorts in the level 3 evaluation to assess the machine learning capability of the synthetic data, and the cohort definitions are provided in **Table 1**. The cohorts were generated for both source and synthetic data. For each cohort, we trained logistic regression using sklearn with the default hyperparameters with 85% of the data, and then tested the model with the remaining data (15%). We reported prevalence/ROC-AUC/PR-AUC in the **Table 2**.

Cohort	Real data	Top P=95%	Top P=100%	Top K=100	Top K=200	TOP K=300
HF readmission	Pre = 25.7 AUC = 65.7 PR = 39.3	Pre = 27.6 AUC = 69.2 PR = 45.7	Pre = 28.4 AUC = 65.9 PR = 41.8	Pre = 30.7 AUC = 68.1 PR = 47.8	Pre = 29.3 AUC = 54.0 PR = 32.9	Pre = 26.5 AUC = 64.9 PR = 39.3
Hospitalization	Pre = 5.6 AUC = 75.3 PR = 19.5	Pre = 5.2 AUC = 77.1 PR = 21.4	Pre = 7.3 AUC = 68.3 PR = 16.5	Pre = 2.8 AUC = 87.0 PR = 22.1	Pre = 5.2 AUC = 84.2 PR = 20.8	Pre = 6.3 AUC = 78.7 PR = 24.6
COPD readmission	Pre = 34.5 AUC = 74.2 PR = 83.8	Pre = 37.8 AUC = 76.4 PR = 84.4	Pre = 47.2 AUC = 74.1 PR = 67.2	Pre = 26.4 AUC = 75.9 PR = 90.3	Pre = 28.3 AUC = 70.1 PR = 82.8	Pre = 34.5 AUC = 68.8 PR = 80.2
Afib ischemic stroke	Pre = 8.7 AUC = 84.0 PR = 48.5	Pre = 10.2 AUC = 78.9 PR = 41.2	Pre = 10.4 AUC = 70.7 PR = 39.1	Pre = 16.6 AUC = 77.1 PR = 50.5	Pre = 15.8 AUC =68.9 PR = 36.6	Pre = 10.8 AUC = 76.8 PR = 38.5
CAD CABG	Pre = 7.1 AUC = 88.4 PR = 55.9	Pre = 4.1 AUC = 81.5 PR = 25.2	Pre = 4.4 AUC = 52.9 PR = 4.3	Pre = 7.2 AUC = 75.6 PR = 38.5	Pre = 4.9 AUC = 73.5 PR = 24.3	Pre = 4.0 AUC = 79.0 PR = 24.1

Table 2 Cohorts prevalence and model performance

Table 2 shows a similar pattern as the level 2 comparison, where the sampling strategies lead to different prevalence and model performance. Encouragingly, the prevalence and model performance of the five cohorts were successfully replicated by one of the sampling strategies. Notably, top p=95% seems to be the best choice for hospitalization, COPD readmission, and afib ischemic stroke, it also has the second-best performance for hf readmission right behind top k=300. As for CAD CABG, top K=100 is the only one that comes close to the metrics associated with real cohort, all the other sampling strategies could not even replicate the prevalence. This suggests that we might need to adjust the sampling strategies depending on the cohorts of interest.

Conclusion

To our knowledge, this is the first attempt to utilize GPT for generating time-series EHR data. Our main contribution lies in the design of a patient representation that captures temporal dependencies among token types, enabling GPT to generate realistic patient sequences. Moreover, this representation allows for easy conversion back to the OMOP format. Our comprehensive evaluation procedures showed that the synthetic data preserved the underlying characteristics of the real patient population.