Large language models as synthetic electronic health record data generators

1st Madhurima Vardhan
Argonne Leadership Computing Facility
Argonne National Laboratory
Lemont, USA
0000-0003-4019-7832

2nd Deepak Nathani
Dept. of Computer Science
University of California, Santa Barbara
Santa Barbara, USA
dnathani@ucsb.edu

3rd Swarnima Vardhan
Dept. of Internal Medicine
Yale New Haven Health System
Bridgeport, USA
swarnima.vardhan@bpthosp.org

4th Abhinav Aggarwal
Dept. of Internal Medicine
Yale University
Bridgeport, USA
abhinav.aggarwal@yale.edu

5th Filippo Simini
Argonne Leadership Computing Facility
Argonne National Laboratory
Lemont, USA
fsimini@anl.gov

***Abstract*—Electronic health record (EHR) data consists of a wealth of information that can be used for driving clinical research and improving patient care. However, due to the complex and sensitive nature of EHR data, there are strict data regulations and privacy concerns around data sharing. Generating adequately validated synthetic EHR data from scratch, such that it is representative of real data, is a viable and attractive solution to address such data-sharing bottlenecks. In this work, we investigate the adoption and implementation of large language models (LLMs) as a sustainable and scalable deep learning approach for generating high-fidelity EHR data. The findings of this study demonstrate that LLMs outperform commonly used generative modeling frameworks, such as variational autoencoders and generative adversarial networks, and recently introduced diffusion models.**

***Index Terms*—Large language models, synthetic data, electronic health record data, generative adversarial network, generative models**

I. Introduction

Recent years have witnessed widespread adoption of electronic health record (EHR) data across medical facilities. Patient EHR datasets are longitudinal, heterogeneous, and encompass thousands of diagnoses and procedural codes. Digital solutions based on EHR datasets have applications across medical treatment planning, patient health monitoring, and designing predictive models for diagnostic care [1], [2]. However, the sensitive nature of patient EHR data precludes data sharing and impedes designing digital solutions due to strict data regulations and privacy concerns. This study addresses the key challenge of ethical and responsible sharing of EHR data by efficiently generating high-quality synthetic longitudinal EHR data using large language models (LLMs).

Conventional methods for sharing EHR data include data de-identification techniques and differential privacy technologies. However, they have challenges associated with designing privacy-loss parameters, and therefore struggle to strike a balance between robust privacy protection and data deployment, creating a gap that synthetic data is uniquely placed to fill [3]. Synthetic data refers to data that is generated from scratch and closely represents real data distributions. Several state-of-the-art generative models have been used for generating longitudinal synthetic EHR data [6], [7], [10].

Existing works on synthetic EHR data generation have been drawn from the field of computer vision, and have focused majorly on applying Generative Adversarial Networks (GANs) [6], [7]. However, training GANs is difficult and they are susceptible to mode collapse. Few recent studies have used Denoising Diffusion Probabilistic Model (DDPM) [10], but DDPM also suffers from generalization challenges. In this work, we build upon the success of LLMs in the field of natural language processing and apply LLMs for generating longitudinal EHR data [12], [13].

While LLMs have demonstrated transformative applications in life science applications such as computational biology, their potential in advancing clinical research, such as for generating synthetic EHR data, remains largely untapped. Modern LLMs are constructed in the format of auto-regressive density models over large sequences of words [12], [13]. **This study proposes to use pre-trained self-attention-based LLMs for probabilistic modeling of longitudinal EHR data.**

Post-processing EHR datasets also involve data transformation and normalization which may result in data loss and introduction of artifacts. Our study addresses this limitation by representing each patient record in the EHR dataset using textual encoding of feature names. Such representation of training data allows for greater contextual information, which we believe can help with superior quality of generated synthetic data using LLMs.

Moreover, in this study we comprehensively evaluate the quality of synthetic EHR data generated using pretrained LLMs by comparing directly with several commonly used generative models such as, variational autoencoders, GANs, and DDPMs using open-source EHR dataset and computing

979-8-3503-5409-6/24/$31.00 ©2024 IEEE
DOI 10.1109/CAI59869.2024.00152

quantitative metrics, such as dimension-wise distribution and covariance. Finally, we also conduct an extensive evaluation to understand the influence of the size of the LLM measured by the number of trainable parameters on the quality of generated EHR data.

II. Methods

In this section, we first describe the formalization of the structure of EHR data. Second, we describe our baseline generative models and their corresponding mathematical notation. Finally, we discuss the evaluation metrics used for comparing the efficiency for generating synthetic EHR data.

A. Electronic Health Record data description

Patient EHR data typically consists of a variety of discrete variables, for example, diagnosis and procedure codes. We use the open-source MIMIC-III Clinical Database, which comprises de-identified EHR data of 46,000 patients collected between 2001 and 2012 [4]. From MIMIC-III data, we extracted disease diagnostic codes, also known as ICD-9 codes. This dataset is used for experiments with binary discrete variables. The extracted data is post-processed to generate a binary matrix in which columns indicate the discrete diagnosis code of each patient in the EHR database. As such, we represent each patient encounter event by a binary vector $x \in \{0,1\}^{|n|}$, assuming there are n discrete variables, where the ith dimension indicates the presence or absence of the ith variable in the patient record. The binary representation resulted in 1071 unique disease diagnostic codes. Therefore, we represent a patient record as a fixed-size vector with 1071 entries for each patient record. This matrix is used as input for training baseline generative models.

B. Baseline Generative Models

1) Variational Autoencoders: A variational autoencoder (VAE) is a type of autoencoder architecture that includes an encoder and a decoder, similar to a standard autoencoder [5]. It is trained to minimize the difference between the original input data and the reconstructed data from the encoded-decoded sequence. The key distinction in a variational autoencoder lies in its approach to the latent space. Unlike standard autoencoders that represent input as a single point in the latent space, a variational autoencoder represents each input as a distribution within this space. To produce a latent variable z such that $z \sim q_{\mu,\sigma}(z) = \mathcal{N}(\mu,\, \sigma^2)\epsilon \sim \mathcal{N}(0,1)$ is sampled, and for a multidimensional vector z is produced by

$$\epsilon \sim \mathcal{N}(0, \mathbf{I})$$

$$\vec{z} \sim \mathcal{N}(\vec{\mu}, \sigma^2\mathbf{I})$$

For both the encoder and the decoder 1D convolutional neural networks were used, each having two hidden layers of size 128. The VAE was trained using the Adam optimizer for 500 epochs and a batch size of 500.

2) Generative Adversarial Networks: Models based on Generative Adversarial Network (GAN) have been a popular choice for generating synthetic EHR data [6]–[8]. Briefly, a GAN model is composed of two neural networks: a generator and a discriminator. These networks are trained together through an iterative process. The generator creates a synthetic sample and the discriminator evaluates whether this sample is real from the training data or fake from the generated synthetic data. The objective of the generator is to produce convincing samples such that the discriminator mistakes them for real data. The generator aims to learn the data distribution p_g over data $\mathbf{x}$ using input noise variables $\mathbf{z}$ from distribution $p_z(\mathbf{z})$. This input noise is transformed by the generator function $G(\mathbf{z};\theta_g)$, where θ_g are the parameters of G, to create the synthetic data. The discriminator, $D(\mathbf{x};\theta_d)$, assesses if its input is real or artificial. It is trained to differentiate between training samples and those produced by G, by minimizing $log(1-D(G(z)))$. Both G and D engage in a min-max game, trying to optimize a value function **F(G,D)**.

$$\min_G \max_D F(G,D) = \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[1 - \log D(G(z))]$$

We evaluate two GAN-based models medGAN and CorGAN in this study that have been demonstrated of generating high-dimensional longitudinal EHR vectors [6], [7]. The medGAN framework uses a pre-trained auto-encoder that alters to a low-dimensional dense space for generation and recovers synthetic EHR using decoders [6]. The CorGAN model is based on using convolutional neural networks to model the autoencoder and the generative network for generating synthetic EHR data instead of multilayer perceptron [7]. All hyperparameters were kept consistent as in the corresponding GitHub repositories: (medGAN, CorGAN).

3) Diffusion Models: More recent works have attempted to use a Denoising Diffusion Probabilistic Model (DDPM) based methods for generating synthetic EHR data [10], [11]. Diffusion models operate through two distinct phases, the forward process and the reverse process [9]. These forward and reverse processes can be described by stochastic differential equations In the forward process, real-world data is progressively distorted by incrementally adding noise, creating training data at various levels of noise for a denoising distribution.

$$d\mathbf{x} = f(\mathbf{x},t)dt + g(t)d\mathbf{w}$$

where x represents data points, w represents standard Wiener process, t is diffusion time and ranges from $\{0$ to $T\}$. The reverse process focuses on producing realistic data by methodically eliminating noise, utilizing the denoising distribution learned from the forward process.

$$d\mathbf{x} = (f(\mathbf{x},t) - g^2(t)\nabla_{\mathbf{x}} log p_t(\mathbf{x}))dt + g(t)d\mathbf{w},$$

where $p_t(\mathbf{x})$ is the marginal density of x at time t and to generate data from random noise, the score function $\nabla_{\mathbf{x}} log p_t(\mathbf{x})$

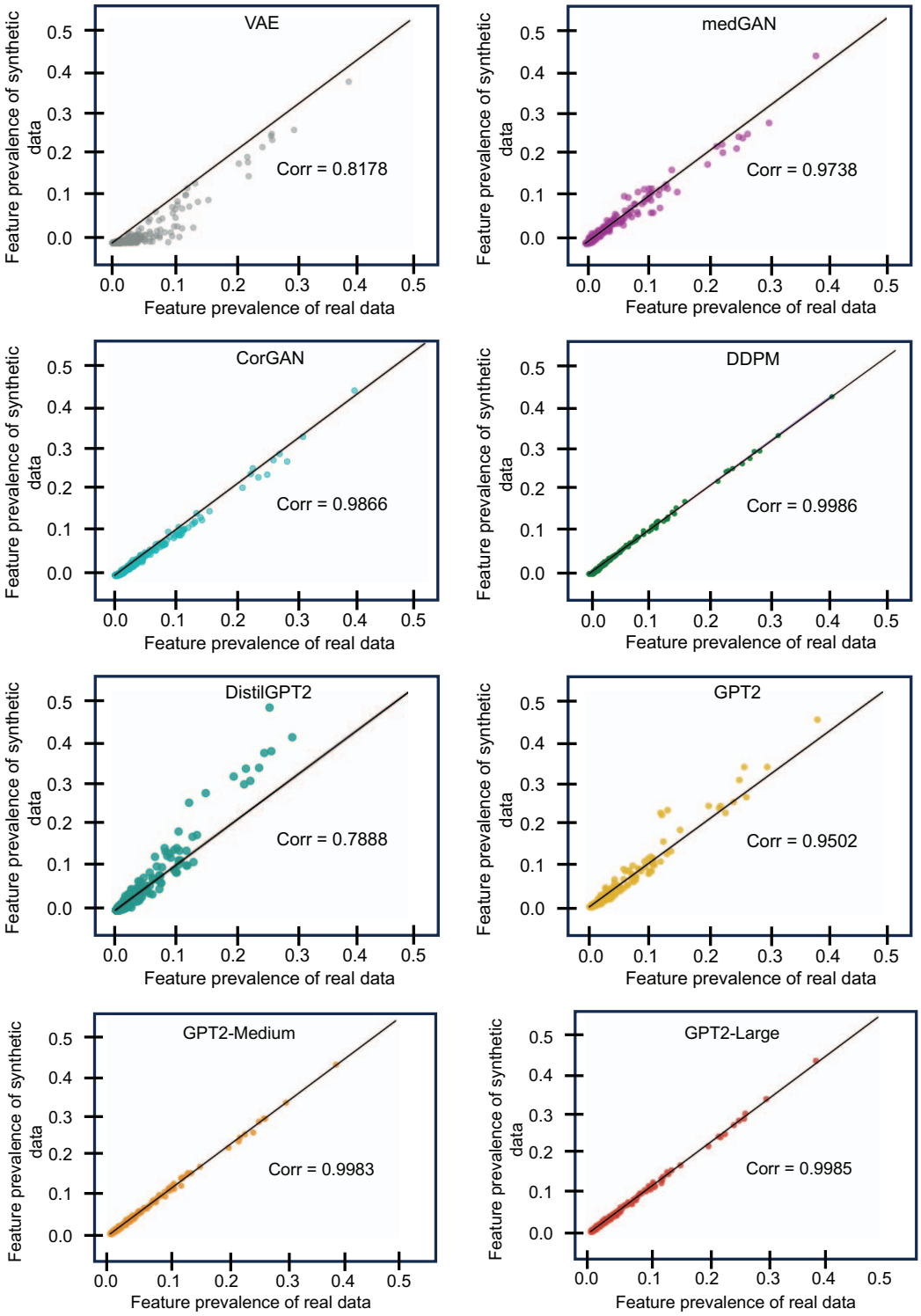

Fig. 1. Distribution of disease diagnostic codes. Dimension-wise distribution to evaluate the quality of generated data, and is measured by the correlation between synthetic and real data. VAE - Variational Autoencoder, GAN - Generative Adversarial Network (medGAN and CorGAN), DDPM - Denoising Diffusion Probabilistic Model, GPT - Generative Pretrained Transformer

is needed to be learned. We compare the model presented by Yuan et al while keeping the hyperparameters consistent as in the ehrdiff repository.

4) Large Language Models: The generative methods described in A-C have been transferred from the computer vision domain, however, in this work, we explore the probabilistic generation of discrete EHR data using LLMs. LLMs are trained to produce a probability distribution for potential subsequent tokens w_k based on an input sequence of arbitrary length $w_1, ..., w_{k-1}$. The probability of natural-language sequences is factorized in an auto-regressive manner in LLMs and is represented as a product of output probabilities conditioned on previously observed tokens [12].

$$p(\mathbf{t}) = p(w_1, ..., wj) = \prod_{k=1}^{j} p(w_k | w_1, ..., w_{k-1}).$$

In particular, we apply self-attention-based LLM, transformer-decoder network architecture such as the Generative Pretrained Transformer (GPT) models, to the EHR data [13], [17]. We represent each row of the tabular EHR data in the format of a sentence which also includes the variable names. Such a representation of EHR data incorporates important contextual information that can be useful when using a pre-trained LLM. To this end, a textual encoder is used that converts each EHR data row with 'Patient ID' and the corresponding 'Diagnostic code' in a sentence with the following structure 'Patient ID X has Diagnostic Code(s)' D_i. We use the framework from [15] to finetune the model on the extracted MIMIC III EHR data, and train the model for 200 epochs to generate synthetic EHR data. A detailed implementation of the LLM modeling framework and hyperparameters can be found in [15] and the corresponding GitHub repository.

Moreover, we assess the influence of LLM size on the quality of generated EHR data. We compare four different pre-trained transformer-decoder LLM models of various sizes [14], [16], [17]. We use the distilled version of GPT-2 that has 82 million parameters as the baseline model, followed by the GPT-2 small version that has approximately 124 million trainable parameters. Furthermore, we evaluate the performance of GPT2-Medium and GPT2-Large models which have 355 and 774 million trainable parameters, respectively.

C. Evaluation metrics

For the experiments conducted in this study, we determine the generative model's performance primarily from the perspective of utility because the objective of the EHR data was to understand whether the study population of the synthetic dataset has a similar distribution as the real EHR data. Utility metrics establish the quality of synthetic EHR, and we use two different utility metrics as described in the following sections.

1) Distribution of disease diagnostic codes: We compute the correlation between real and synthetic EHR data for evaluating the distribution of disease diagnostic codes by computing the dimension-wise distribution. The concept of dimension-wise distribution refers to the similarity in features between synthetic and real data. This metric is used to assess whether a generative model can accurately replicate the complex, high-dimensional distribution of real EHR data. For each code dimension, the empirical mean is computed separately for both synthetic and real EHR data. This mean represents how frequently each code occurs, or its prevalence. To illustrate the dimension-wise distribution, scatter plots are used, where each axis denotes the prevalence of codes in synthetic and real EHR data, respectively. Higher correlation numbers correspond to better quality of generated data that more closely represents real EHR data distribution patterns.

2) Covariance of disease diagnostic codes: The covariance of two disease diagnostic codes is a measure of co-occurrence of the two disease codes in a population. Specifically, if X_i is a binary random variable denoting the presence ($X_i = 1$) or absence ($X_i = 0$) of disease diagnostic code i, then $E[X_i]$ is the frequency of disease diagnostic code i in a population (i.e. the fraction of individuals with disease code i), $\mathrm{E}[X_i X_j]$ is the frequency of co-occurrence of the two disease codes i and j in a population (i.e. the fraction of individuals with both diseases i and j) and

$$\begin{aligned}\operatorname{cov}(X_i, X_j) &= \mathrm{E}\left[(X_i - \mathrm{E}[X_i])(X_j - \mathrm{E}[X_j])\right] \\ &= \mathrm{E}[X_i X_j] - \mathrm{E}[X_i]\,\mathrm{E}[X_j]\end{aligned}$$

is the difference between the observed frequency of co-occurrence and the one that would hypothetically be obtained by removing any association between the two diseases (i.e. by randomly reassigning the disease codes among individuals). In particular, $\operatorname{cov}(X_i, X_j) = 0$ if there is no non-random association between disease codes i and j, $\operatorname{cov}(X_i, X_j) > 0$ if the co-occurrence of the two disease codes is higher than what is expected by chance (i.e. there is a positive association: a individual with disease code i will likely have disease code j as well), and $\operatorname{cov}(X_i, X_j) < 0$ if the co-occurrence of the two disease codes is smaller than what is expected by chance (i.e. there is a negative association: a individual with disease code i will likely not have disease code j). The covariance matrix C with elements $C_{ij} = \operatorname{cov}(X_i, X_j)$ contains the covariances of all pairs of disease diagnostic codes and includes the information about their co-occurrences.

III. Results and Discussion

In this section, we report the evaluation results of different LLM models to generate synthetic EHR data and the influence of LLM model size, measured with respect to trainable parameters. Moreover, to demonstrate the effectiveness of LLMs, we compare the performance of LLMs with several baseline generative models: 1) VAE 2) GAN models (medGAN and CorGAN) 3) DDPM by interrogating the dimension-wise distributions and covariances of generated data of each model. For all baseline models, we divide the binary matrix, which is extracted from the MIMIC III raw data as described in Section II-A, into a training $D_{training} \in \{0,1\}^{RX|M|}$ and a test set $D_{test} \in \{0,1\}^{TX|M|}$, where $|M|$ is the feature

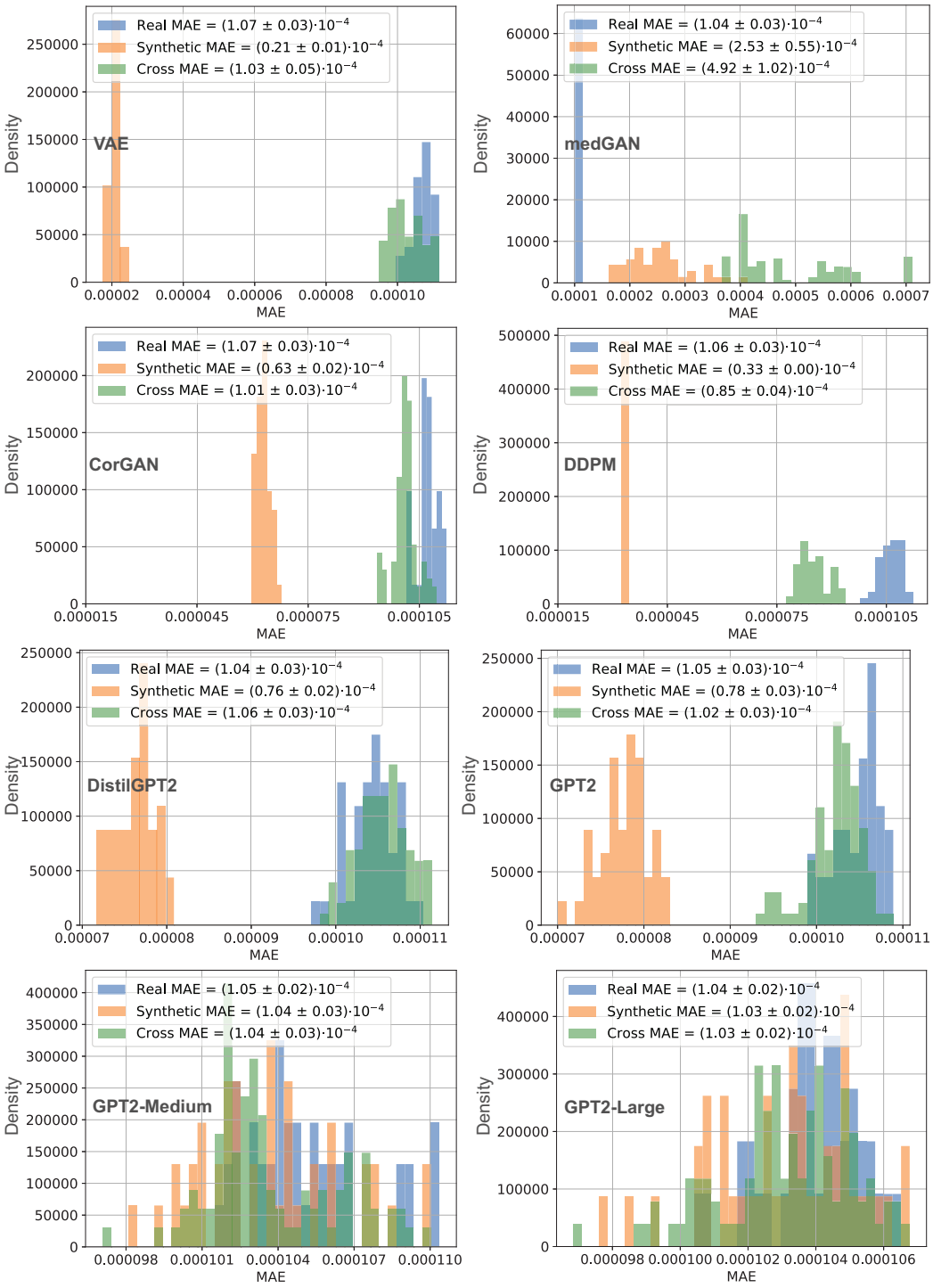

Fig. 2. Covariance of disease diagnostic codes. Column-wise correlation to evaluate the quality of generated data, and is measured by the closeness between the three histograms representing MAE^{real}, the MAE distance between all pairs of covariance matrices of the real groups, MAE^{synt}, the MAE distance between all pairs of covariance matrices of the synthetic groups, and MAE^{cross}, the cross MAE distance between a real covariance matrix and a synthetic covariance matrix, for all pairs. . VAE - Variational Autoencoder, GAN - Generative Adversarial Network (medGAN and CorGAN), DDPM - Denoising Diffusion Probabilistic Model, GPT - Generative Pretrained Transformer

size. The size of the training set and test set is consistent for all baseline models. We use $D_{training}$ to train the models, then generate synthetic samples $D_{synthetic} \in \{0,1\}^{SX|M|}$ using the trained model. For different LLM models, DistilGPT2, GPT2, GPT2-Medium and GPT2-Large, we finetune the model using the textually encoded raw MIMIC III data and generate synthetic EHR data. Therefore to compare with baseline models, the LLM-generated synthetic EHR data is post-processed in the binary matrix format consisting of 1071 unique codes as columns for each patient record. The number of samples for $D_{synthetic}$ and $D_{training}$ are kept consistent for all evaluations.

A. Distribution of disease diagnostic codes

The key metric for high-quality synthetic EHR data is such that it establishes that the generated data learns the distribution of the real data (for each dimension, i.e. 1071 unique diagnostic codes). Therefore, to demonstrate this evaluation we report the dimension-wise probability because this measurement refers to the Bernoulli success probability of each disease diagnostic code. These results are presented in Figure 1 and as can be seen, the GPT2-Medium and GPT2-Large model are superior compared to VAE and both GAN-based methods (medGAN and CorGAN) and equivalent to DDPM. The VAE model does not generate data when the probability of occurrence of a disease diagnosis code is higher than the corresponding occurrence in the real EHR data. While medGAN and CorGAN also show impressive correlations ($Corr = 0.9738$ to $Corr = 0.9866$), these methods make use of an autoencoder, which is difficult to train and is prone to mode collapse. Despite the high $Corr = 0.9986$, DDPM suffers from the generalization challenge, especially in generating data away from the modes. While generating high-quality synthetic EHR data, pre-trained LLMs such as those used in this study can be efficiently fine-tuned for a specific task. Furthermore, by virtue of textual encoding, these models are extensible and generalizable for heterogeneous data.

Among different LLMs, Figure 1 demonstrates that the model size, determined based on the number of trainable parameters, has a strong influence on the quality of generated synthetic EHR data. The quality of synthetic EHR data significantly improves ($Corr = 0.7888$ to $Corr = 0.9502$) with an increase in the number of trainable parameters from DistilGPT2 to GPT2. Similarly increasing the size of the model further from GPT2 to GPT2-Medium, furthermore improves the correlation ($Corr = 0.9502$ to $Corr = 0.9983$). Interestingly, correlation only marginally improve from GPT2-Medium to GPT2-Large ($Corr = 0.9983$ to $Corr = 0.9985$).

B. Covariance of disease diagnostic codes

In order to assess whether the synthetic data has the same co-occurrence patterns as the real data, we compute the covariance matrices of the real data C^{real} and of the synthetic data C^{synt} and measure their distance as the Mean Absolute Error MAE $= \mathrm{E}[|C^{real} - C^{synt}|]$. We use the following resampling technique to assess how closely the synthetic data approximates the real data: First, we separately assign real and synthetic individuals to $n = 10$ groups at random, then we compute the covariance matrix C of each group, real and synthetic. Finally, we compute three histograms for the following quantities: MAE^{real}, the MAE distance between all pairs of covariance matrices of the real groups, MAE^{synt}, the MAE distance between all pairs of covariance matrices of the synthetic groups, and MAE^{cross}, the cross MAE distance between a real covariance matrix and a synthetic covariance matrix, for all pairs. If the synthetic covariances are indistinguishable from the real ones, the three histograms should overlap. In Figure 2, maximum overlap can be seen for GPT2-Medium and GPT2-Large models. Therefore synthetic EHR data generated using these models exhibit the same co-occurrence patterns as the real EHR data. This result implies that the quality of generated EHR data using pretrained LLMs is superior compared to all baseline models (VAE, GAN and DDPM).

IV. Conclusion

EHR data contains critical and exhaustive information about patient health. The widespread clinical use of EHR data will play key role in designing novel digital solutions for advancing evidence-based medicine. However, the sensitive nature of EHR data impedes upon its applications due to stringent data-sharing protocols. Therefore, this study explored LLMs for synthetic EHR data generation because synthetic data that maintains the statistical properties of real data is a viable and scalable solution for preserving privacy without the vulnerability to risks of data leakage. This work establishes that pre-trained LLMs can reliably and effectively generate high-quality synthetic EHR data, and are superior to state-of-art generative models such as VAE, GANs, and DDPM. Moreover, the ease of fine-tuning LLMs for a specific task and textual encoding of tabular data enables generalization to heterogeneous and longitudinal EHR data.

References

[1] Shang, J., Xiao, C., Ma, T., Li, H. and Sun, J. Gamenet: Graph augmented memory networks for recommending medication combination. In proceedings of the AAAI Conference, vol. 33, 1126–1133 (2019).

[2] Farrar, C. R. & Worden, K. Structural health monitoring: a machine learning perspective (John Wiley & Sons, 2012).

[3] Garfinkel, Simson L., John M. Abowd, and Sarah Powazek. "Issues encountered deploying differential privacy." In Proceedings of the 2018 Workshop on Privacy in the Electronic Society, pp. 133-137. 2018.

[4] Pollard, Tom J., and A. E. W. Johnson III. "The MIMIC III Clinical Database." The MIMIC-III Clinical Database. PhysioNet (2016).

[5] Kingma, Diederik P., and Max Welling. "Auto-encoding variational bayes." arXiv preprint arXiv:1312.6114 (2013).

[6] Choi, Edward, Siddharth Biswal, Bradley Malin, Walter F. Stewart, and Jimeng Sun. "Generating multi-label discrete patient records using generative adversarial networks." MLHC, pp. 286-305. PMLR, 2017.

[7] Torfi, Amirsina, and Edward A. Fox. "CorGAN: correlation-capturing convolutional generative adversarial networks for generating synthetic healthcare records." arXiv preprint arXiv:2001.09346 (2020).

[8] Fang, Mei Ling, Devendra Singh Dhami, and Kristian Kersting. "Dp-ctgan: Differentially private medical data generation using ctgans." In Int. Conference on AI in Medicine, pp. 178-188, 2022.

[9] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020.

[10] Yuan, Hongyi, Songchi Zhou, and Sheng Yu. ”EHRDiff: Exploring Realistic EHR Synthesis with Diffusion Models.” arXiv preprint arXiv:2303.05656 (2023).
[11] Tian, Muhang, Bernie Chen, Allan Guo, Shiyi Jiang, and Anru R. Zhang. ”Fast and Reliable Generation of EHR Time Series via Diffusion Models.” arXiv preprint arXiv:2310.15290 (2023).
[12] Yoshua Bengio, Réjean Ducharme,and Pascal Vincent. ”A neural probabilistic language model”. Advances in neural information processing systems (NeurIPS), 13, 2000.
[13] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Uszkoreit, Llion Jones, Aidan Gomez, Łukasz Kaiser, and Illia Polosukhin. ”Attention is all you need.” Advances in neural information processing systems (2017).
[14] Budzianowski, Paweł, and Ivan Vulić. ”Hello, it’s GPT-2–how can I help you? towards the use of pretrained language models for task-oriented dialogue systems.” arXiv preprint arXiv:1907.05774 (2019).
[15] Borisov, Vadim, Kathrin Seßler, Tobias Leemann, Martin Pawelczyk, and Gjergji Kasneci. ”Language models are realistic tabular data generators.” arXiv preprint arXiv:2210.06280 (2022).
[16] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. In NeurIPS EMC2 Workshop, 2019.
[17] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. Advances in neural information processing systems, 33:1877–1901, 2020.