Continuous Patient-Centric Sequence Generation via Sequentially Coupled Adversarial Learning

Lu Wang, Wei Zhang(✉), and Xiaofeng He(✉)

School of Computer Science and Software Engineering,
East China Normal University, Shanghai, China
joywanglulu@163.com, zhangwei.thu2011@gmail.com, xfhe@sei.ecnu.edu.cn

Abstract. Analyzing massive patient-centric Electronic Health Records (EHRs) becomes a key to success for improving health care and treatment. However, the amount of these data is limited and the access to EHRs is difficult due to the issue of patient privacy. Thus high quality synthetic EHRs data is necessary to alleviate these issues. In this paper, we propose a Sequentially Coupled Generative Adversarial Network (SC-GAN) to generate continuous patient-centric data, including patient state and medication dosage data. SC-GAN consists of two generators which coordinate the generation of patient state and medication dosage in a unified model, revealing the clinical fact that the generation of patient state and medication dosage data have noticeable mutual influence on each other. To verify the quality of the synthetic data, we conduct comprehensive experiments to employ these data on real medical tasks, showing that data generated from SC-GAN leads to better performance than the data from other generative models.

Keywords: Continuous data · Patient-centric sequence · Sequentially coupled adversarial learning

1 Introduction

The effective analysis of Electronic Health Records (EHRs) has the potential to improve clinical outcomes. However, since data of EHRs largely consists of personal medical information, it raises a significant privacy issue which discourages the public sharing of these data. In addition, the amount of these data is limited, because most of EHRs are self-governed by healthcare organizations which require formal collaborations and complex data usage agreements for even academic research purpose. Thus, the limited access to EHRs becomes the bottlenecks of advancing the field of healthcare [1] and hinders the development of medical data mining solutions.

Simulation is a standard practice for medical data generation and learning. Due to the complex hand-crafted rules of simulation design, automatically

G. Li et al. (Eds.): DASFAA 2019, LNCS 11447, pp. 36–52, 2019.
https://doi.org/10.1007/978-3-030-18579-4_3

generating synthetic data becomes a fashion for relieving privacy risks and the data scarcity issue [2,3]. Specifically, deep generative models have recently been employed for releasing medical data mining [4,5]. The generated medical data can be exploited for mitigating the risk of privacy and alleviate the data scarcity issue by data augmentation.

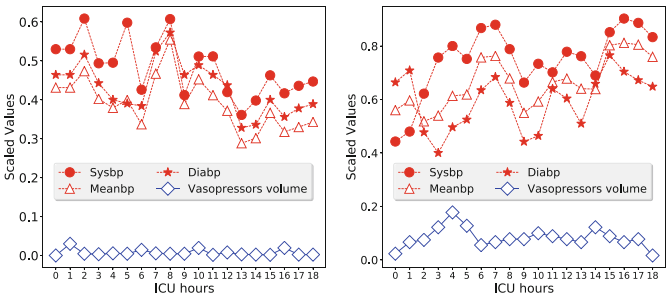


Fig. 1. The records in MIMIC-III show three blood pressure measurements of two sepsis patients and the vasopressors dosage prescribed to them, where vasopressors is used to counteract sepsis-induced vasodilation and elevate arterial pressure. The left patient who takes small dosage of vasopressors shows a declining blood pressure. The right patient who takes larger dosage has a rising blood pressure.

Generative Adversarial Networks (GANs) [6] train a generative model G and a discriminative model D simultaneously with antagonistic objectives which achieves promising results in generating realistic samples such as images [7–9], text [10,11], etc. Only recently, a very few studies [4,5,12] apply GAN to synthesize medical data generation. However, [4,5] focus on generating patient state data, ignoring medication dosage data which is another crucial type of patient-centric data. Although [12] enables the simultaneous generation of patient state and corresponding medications, it could only generate discrete values at a specified time, unable to generate continuous sequential medical data which is more in line with reality.

In this paper, we focus on the generation of continuous patient-centric sequence, mainly including patient state and corresponding medication dosage data, both of which play an important role in treatment recommendation [13,14]. The key observation is that the generation of patient state and medication dosage data have significant mutual influence on each other. On one hand, doctors determine medication dosage mainly based on patients' current state, leading to the generation of medication dosage influenced by the generation of patient state. On the other hand, the state of patients highly depend on the medication dosage they take. For example, various fluids and vasopressor dosage strategies have been proved to cause extreme variations in patient [15]. As shown in Fig. 1, the blood pressures of sepsis patients are affected by the dosage of vasopressor they take and the doctor also adjusts vasopressor dosage according to the blood pressures.

Inspired by the above observation, we propose a Sequentially Coupled GAN (SC-GAN) model to generate the state of patients and medication dosage together, which captures the interaction between them. Specifically, SC-GAN consists of coupled generators: one is leveraged to first generate current state of patients and the other further utilizes the acquired state to generate the corresponding medication dosage prescribed to each patient. As a result, the two generators are directly associated and trained jointly in a unified model to benefit each other.

Our main contributions are summarized as follows:

- We propose SC-GAN which consists of two interacted generators to produce both the state of patients and the corresponding medication dosage. The coupled generators capture the mutual influence of their generation, which is overlooked by previous studies. In addition, we adopt a hybrid loss trick which combines feature matching loss and standard generator loss to further improve the performance.
- Experiments on public available real-world EHRs show the treatment recommendation model trained on the synthetic data generated by our model achieves better performance than state-of-arts and incorporating the synthetic data into the real datasets can further improve the performance.

2 Related Work

In this section, we overview the related studies from two aspects: sequentially generative adversarial networks and medical data generation.

2.1 Sequentially Generative Adversarial Networks

Generative adversarial networks are generative models with the mechanism of adversarial training, where the goal of D is to discriminate between real data and the samples generated by G, and the goal of G is to fool D with generated realistic data. Although GANs have achieved impressive success in image generation [16], there are limited studies using GANs to produce sequential data. The most conventional methods with this regard are recurrent neural networks (RNNs). RNNs have been utilized to generate sequential discrete tokens (e.g., machine translation [17]) and continuous values (e.g., music data [18,19]). The most common objective for optimizing RNNs is based on maximum likelihood. However, utilizing this criterion to generate sequence data has been argued to suffer from the *exposure bias* [20]. In contrast, GANs work well to mitigate this problem. SeqGAN [10] extends GAN with RNN to generate sequences of discrete word tokens via policy gradient. C-RNN-GAN [21] trains RNNs with adversarial training for continuous music generation, which is a pioneering study to generate sequential and continuous data. Several methods also employ convolutional neural networks (CNNs) for generating audios and images. Although using convolutional GANs to generate sequential data may have faster training speed than recurrent GANs, it loses the Markov property of trajectory samples.

2.2 Medical Data Generation

The generated medical data helps to build predictive systems in the medical domain, such as predicting the patient-specific trajectories (e.g., Albumin, Arterial pH, Calcium, etc.) or recommending treatments for a given patient. Although the most commonly acceptable approach to generate EHRs dataset for sharing is de-identification [22], the individual information of the patients can be re-identified through residual distinguishable patterns [12]. For example, re-identifying lab tests, demographics, and genomic variants. Generating synthetic data becomes an alternative approach to reduce the privacy risk.

Followed by the successful applications of GANs mentioned above, a set of studies begin to employ GANs to generate medical data for sharing. Li et al. [23] proposed a hybrid GAN to generate text reports for medical image with high-level and low-level modules. Most related to this work, Yahi et al. [5] utilized RNNs with adversarial learning to generate the laboratory test time series data. Nevertheless, it overlooks the fact that the patients' state are highly influenced by the medications they take. On the other hand, Beaulieu et al. [4] employed the Auxiliary Classifier Generative Adversarial Network (AC-GAN) [9] to generate real-valued state of patient to provide a freely accessible public version for discovery-oriented analysis. Esteban et al. [24] used a recurrent GAN to generate real-valued time-series state of patients, which also only considered the generation of state as [9]. Edward et al. [12] combined autoencoder with GAN to generate the discrete variables such as diagnosis, medication and procedure codes. However, it only considers one-step generation instead of sequential generation.

To reflect the clinical fact, we propose SC-GAN to capture the interactions between continuous state of patients and the medication dosage they take. Specifically, SC-GAN designs coupled generators to produce the interdependent state and medication dosage. Note that our proposed model is significantly different from the Multi-Generator generative adversarial net [25] which utilizes multiple generators to generate one single type of data.

3 Preliminaries

In this section, we first briefly introduce the data of continuous patient-centric sequence and some basic notations, followed by the problem definition and the description of GANs with its main variants.

3.1 Data Description and Notations

The major goal of this paper is to generate patient-centric medical sequence data. Unlike previous studies [4,5,12], we focus on generating continuous values with consideration of their mutual interactions. Generally, the numerical data could be categorized into two aspects: (1) patient state data which includes lab tests, vital signs, intake, etc; (2) medication dosage data.

Since both patient state data and medication dosage data present sequential characteristics, we assume $\mathbf{s}_t$ denotes the state data of a patient at time step t. Noting that to simplify the notations, we omit the subscript about a specific patient and can apply these notations to different patients. Correspondingly, we represent the medication dosage data as $\mathbf{a}_t$. For example, in the clinical practice, doctors always design proper medication dosage for patients based on their current state [13,26]. Meanwhile, patient state would vary after taking the medication dosage.

Based on this intuition, we implement SC-GAN by ensuring that the current state data $\mathbf{s}_t$ is generated based on the previous state data $\mathbf{s}_{t-1}$ and the medication dosage data $\mathbf{a}_{t-1}$, while in turn the medication dosage data $\mathbf{a}_t$ is produced based on the input of the current state $\mathbf{s}_t$.

3.2 Problem Definition

The detail of the problem studied in this paper is described in Problem 1.

Problem 1 [**Continuous Patient-centric Sequence Generation**]. For a patient with given disease and medication, we aim at generating the sequential state $\mathbf{S} = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_T\}$ and corresponding dosage $\mathbf{A} = \{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_T\}$ with the consideration of their mutual interactions.

3.3 Basics of GANs

Generative Adversarial Networks are generative models which consist of two neural networks: Discriminator Net $D(\mathbf{x}; \theta_d)$ and Generator Net $G(\mathbf{z}; \theta_g)$, where $\mathbf{z}$ is a random noise. $D(\mathbf{x})$ indicates the probability that $\mathbf{x}$ comes from a real data distribution. To discriminate the real data from synthetic data, it maximizes the probability of real data and minimizes the probability of synthetic data generated from G. In contrast, G has the opposite goal which is to generate realistic data to make D indistinguishable. That is, D and G play the following minimax game to reach a Nash equilibrium:

$$\begin{aligned} min_G max_D V(D,G) = \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[logD(\mathbf{x})] \\ + \mathbb{E}_{\mathbf{z}\sim p_z(\mathbf{z})}[log(1 - D(G(\mathbf{z})))] \end{aligned} \tag{1}$$

where p_{data} represents the distribution of real data. $p_z(\mathbf{z})$ denotes the distribution of noises where normal distribution $\mathcal{N}(0,1)$ and uniform distribution $\mathcal{U}(0,1)$ are the common choices. D and G are iteratively optimized.

Unfortunately, if the discriminator D is excessively strong, then Eq. 1 may be unable to give sufficient gradient for G to update its parameters. Instead of using the objective which maximizes the predicted probability of the discriminator, Salimans et al. proposed feature matching [27] which is a new strategy to optimize G.

Feature matching aims at generating samples actually fall into the real data manifold. It also encourages greater variance in G as well as prevents it from

overtraining on the current discriminator. Formally, the feature matching loss is described as follows:

$$L_G = \|\mathbb{E}_{x \sim p_{data}} \mathbf{f}(\mathbf{x}) - \mathbb{E}_{z \sim p_z(\mathbf{z})} \mathbf{f}(G(\mathbf{z}))\|_2^2 \tag{2}$$

where $\mathbf{f}(\mathbf{x})$ is the representation of the last layer before the final classification of D.

Inspired by the above methodologies, we provide our SC-GAN model by coupling two sequential generators in a unified generative adversarial network, capturing the mutual interaction in continuous patient-centric medical sequence.

4 Methodology

4.1 Overview of SC-GAN

SC-GAN aims at generating synthetic patient-centric medical data which consists of trajectory state data $\mathbf{S}$ ($\mathbf{S} = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_T\}$) of patients and corresponding medication dosage $\mathbf{A}$ ($\mathbf{A} = \{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_T\}$) during treatment process. Specifically, the medication dosage data $\mathbf{a}_t$ at time step t is generated based on the current state data $\mathbf{s}_t$ of a patient, and the current state $\mathbf{s}_t$ is generated based on the previous state data $\mathbf{s}_{t-1}$ and previous medication dosage data $\mathbf{a}_{t-1}$. As shown in Fig. 2, we establish two generators G_1 and G_2 for generating $\mathbf{S}$ and $\mathbf{A}$ respectively, where $\mathbf{a}_t$ is with the input of random noise $\hat{\mathbf{z}}_t^a$, state $\mathbf{s}_t$, and $\mathbf{s}_t$ is with the input of random noise $\hat{\mathbf{z}}_{t-1}^s$, $\mathbf{a}_{t-1}$ and $\mathbf{s}_{t-1}$. In other words, the two generators interact with each other to generate these two categories of data. To implement the discriminator, we set a classification task to distinguish the real and synthetic data at each time step.

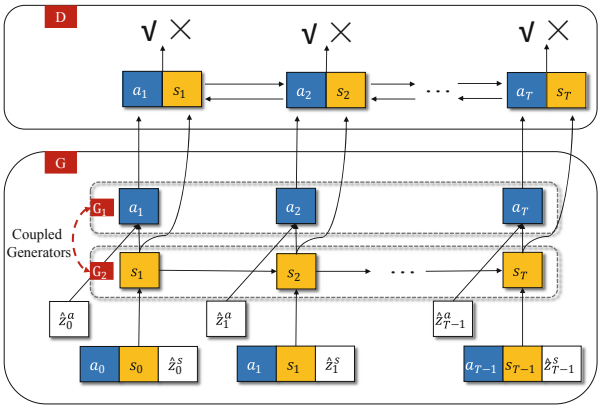


Fig. 2. The general framework of Sequentially Coupled GAN. This model consists of three main components: a discriminator (a 2-layer bidirectional LSTM) and two interdependent generators (two 2-layer LSTMs). $\mathbf{s}_t$ indicates trajectory state of the patients at time-point t. $\mathbf{a}_t$ represents the medication dosage used for patients. $\hat{\mathbf{z}}_t^a$ and $\hat{\mathbf{z}}_t^s$ are random noises of the medication dosage and patient state respectively. The correct symbol (✓) indicates the discriminator determines the data is real while the incorrect symbol (×) indicates the data is judged as synthetic.

4.2 Coupled Generators

The goal of the coupled generators G_1 and G_2 is to generate realistic synthetic medical records of patients to maximize the probability of D for letting D make a misjudgment. Both G_1 and G_2 have two layers of long short-term memory networks (LSTM) [28]. The reasons are that: (1) LSTM is capable of exhibiting temporal dynamics compared to feed-forward networks and CNNs; (2) LSTM utilizes three gates to protect and control the cell state, which mitigates the gradient vanishing and exploding problems compared to RNNs.

G_1 generates medicaiton dosage data $(\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_T)$ with the input of sequential continuous state data $(\mathbf{s}_0, \mathbf{s}_1, ..., \mathbf{s}_{T-1})$ and a random noise sequence $(\hat{\mathbf{z}}_0^a, \hat{\mathbf{z}}_1^a, ..., \hat{\mathbf{z}}_{T-1}^a)$. Formally, at each time step t, the input $\mathbf{z}_t^a$ of G_1 is the concatenation of $\mathbf{s}_t$ and $\hat{\mathbf{z}}_t^a$:

$$\mathbf{z}_t^a \leftarrow [\mathbf{s}_t; \hat{\mathbf{z}}_t^a], \tag{3}$$

$$\mathbf{a}_t = G_1(\mathbf{z}_t^a), \tag{4}$$

where $\mathbf{s}_t$ is the output of G_2 at time step t and $\hat{\mathbf{z}}_t^a \in \mathcal{U}(0, 1)$.

G_2 is leveraged to generate the patient state data $\mathbf{s}_t$ with the input of previous state $\mathbf{s}_{t-1}$, the medication dosage data $\mathbf{a}_{t-1}$, and the current random noise $\hat{\mathbf{z}}_t^s$. In other words, at each time step of G_2, the input $\mathbf{z}_t^s$ at time step t is the concatenation of $\mathbf{s}_{t-1}$, $\mathbf{a}_{t-1}$, and $\hat{\mathbf{z}}_t^s$:

$$\mathbf{z}_t^s \leftarrow [\mathbf{s}_{t-1}; \mathbf{a}_{t-1}; \hat{\mathbf{z}}_t^s], \tag{5}$$

$$\mathbf{s}_{t-1} = G_2(\mathbf{z}_{t-1}^s), \mathbf{a}_{t-1} = G_1(\mathbf{z}_{t-1}^a), \tag{6}$$

where $\mathbf{s}_{t-1}$ is the output of G_2 at time step $t-1$, $\mathbf{a}_{t-1}$ is the output of G_1 at time step $t-1$, and $\hat{\mathbf{z}}_t^a$ is also a uniform random value in [0,1].

As shown in Eqs. 4 and 6, the outputs of G_1 and G_2 are also the inputs of G_2 and G_1. Combining these two generators together and differentiating different patients, we minimize the following objective function to train these generators:

$$L_G = \frac{1}{N}\frac{1}{T}\sum_{i=1}^{N}\sum_{t=1}^{T} log(1 - D(G(\mathbf{z}_{i,t}))) \tag{7}$$

$$G(\mathbf{z}_{i,t}) = [G_1(\mathbf{z}_{i,t}^a); G_2(\mathbf{z}_{i,t}^s)] \tag{8}$$

where N is the number of patients, T is the time length of the patient record. That is, the optimal generator G should generate both realistic patient state data and medication dosage data at the same time.

As for training, we first conduct a supervised pretraining step for SC-GAN. To avoid training an excessively strong discriminator to hamper the generators optimization, we only pretrain the coupled generators by utilizing the least square loss to generate the sample of the next time step. The objective function for the pretraining step is defined as follows:

$$\begin{aligned} L_{pretrain} = \frac{1}{N}\frac{1}{T}\sum_{i=1}^{N}\sum_{t=1}^{T}(&\|\mathbf{s}_{i,t}^{real} - G_1(\mathbf{z}_{i,t}^a)\|_2^2 \\ &+ \|\mathbf{a}_{i,t}^{real} - G_2(\mathbf{z}_{i,t}^s)\|_2^2) \end{aligned} \tag{9}$$

where $\mathbf{s}_{i,t}^{real}$ and $\mathbf{a}_{i,t}^{real}$ indicate the real state and dosage data of patient i at time step t, respectively. In the pretraining step, the coupled generators are with the input of concatenation of the random noise and the real data from training set.

During adversarial training, the goal of generator G is to produce samples which can cheat D. Thus G could map $p_z(\mathbf{z})$ to only a few and low-volume regions. That is, G may produce the same synthetic data. This problem is called *mode collapse*. To improve the variance of G and address the instability of GANs, we combine the feature matching loss shown in Eq. 2 with the standard generator loss through a weighted linear fusion, which is defined as follows:

$$\begin{aligned} L_G = \frac{1}{N}\frac{1}{T}\sum_{i=1}^{N}\sum_{t=1}^{T}\Big(\lambda_{fm}\big(\|\mathbf{f}(\mathbf{s}_{i,t}^{true}) - \mathbf{f}(G_1(\mathbf{z}_{i,t}^{a}))\|_2^2 \\ + \|\mathbf{f}(\mathbf{a}_{i,t}^{true}) - \mathbf{f}(G_2(\mathbf{z}_{i,t}^{s}))\|_2^2\big) \\ + \lambda_{adv}\big(log(1 - D(G(\mathbf{z}_{i,t})))\big)\Big) \end{aligned} \tag{10}$$

where $\lambda_{fm} \in [0, 1]$ and $\lambda_{adv} \in [0, 1]$ are the weights of feature matching loss and standard generator loss, respectively, and they are tuned empirically based on the test performance. $\mathbf{f}$ is the representation of the last layer before final classification of D.

4.3 Discriminator

The goal of discriminator $D(\mathbf{x})$ is to correctly judge the real data and the generated synthetic data. SC-GAN classifies the data into real or synthetic at each time step, to simplify the procedure of directly discriminating the whole sequence of data. Specifically, D is trained to minimize the negative cross-entropy loss between the real sequential patient-centric records $([\mathbf{s}_1; \mathbf{a}_1], [\mathbf{s}_2; \mathbf{a}_2], ..., [\mathbf{s}_T; \mathbf{a}_T])$ and the generated data $(G(\mathbf{z}_1), G(\mathbf{z}_2), ..., G(\mathbf{z}_T))$. D has a 2-layer bidirectional LSTM, which could integrate the context in both directions. Finally, the loss of the discriminator can be described as follows:

$$L_D = -\frac{1}{N}\frac{1}{T}\sum_{i=1}^{N}\sum_{t=1}^{T}\Big(logD\big(\mathbf{x}_{i,t}\big) + log\big(1 - D(G(\mathbf{z}_{i,t}))\big)\Big) \tag{11}$$

$$\mathbf{x}_{i,t} = [\mathbf{s}_t; \mathbf{a}_t], G(\mathbf{z}_{i,t}) = [G_1(\mathbf{z}_{i,t}^{a}); G_2(\mathbf{z}_{i,t}^{s})] \tag{12}$$

where $\mathbf{x}_{i,t}$ consists of the real state of patient i and his/her medication dosage data at time step t.

5 Experiments

In this section, we conduct extensive experiments on two distinct patient-centric datasets extracted from real-world EHRs, aiming at demonstrating the data generated by SC-GAN is better than those of several comparative models. The source code will be released with the publication of this paper for relevant study.

5.1 Dataset Description and Preprocessing

Patient Cohort. The experiments are conducted on a real-world EHRs, namely the Multiparameter Intelligent Monitoring in Intensive Care (MIMIC-III v1.4) database [29]. MIMIC-III encompasses a population of 43 K patients and 474 million patient-centric state observations in intensive care units (ICUs) during 2001 and 2012. Based on MIMIC-III, we construct two distinct disease datasets: sepsis and diabetes. Sepsis is a main cause of mortality in ICUs [13] and a great deal of studies try to find optimal treatment dosage for sepsis. Diabetes is a lifestyle-related chronic disease and glycemic control with proper dosage is essential for diabetes [30].

We extract sepsis patients conforming to the Sepsis-3 criteria [31] and extract diabetes, mycosis and isoniazid patients with ICD-9 codes for diabetes. We summarize the basic statistics of the extracted patients in Table 1. We randomly divide the dataset for training, validation, and testing sets by the proportion of 80:10:10.

Table 1. Basic statistics of the two datasets

Description	Sepsis-3	Diabetes	Mycosis	Isoniazid
`% Female`	43.6	32.1	44.8	41.9
`Mean age`	66.6	76.8	62.7	68.2
`Hours in ICU`	59.3	82.7	63.4	79.5
`Total population`	13,773	5,538	6,722	3,245

State of Patients. For each patient, we extract relevant physiological parameters as his/her state, such as laboratory tests, vital signs and output events. The details of these features are shown in Table 2. We aggregate the data into windows of 4 h to obtain patient-centric multidimensional time series data. The missing variables are imputed by k-nearest neighbors and the records with more than 10 missing variables are removed. We rescale each feature at each time step independently to the range [0, 1].

Medication Dosage of Patients. We select intravenous fluids (IV fluids) and vasopressor as the main medications of sepsis patients and choose insulin for diabetes patient. Because the dosage of these medications have been verified to highly affect the state of diabetes patients and sepsis patients, and even their mortality in ICU.

Table 2. Description of the trajectory state of patients.

Laboratory tests	Albumin, Arterial_pH, Calcium, Glucose, Partial Thromboplastin Time, Potassium, SGPT, Arterial Blood Gas, Blood Urea Nitrogen, Chloride, International Normalized Ratio, Sodium, Ionised Calcium, Arterial Lactate, CO2, Creatinine, Prothrombin Time, SGOT Platelets Count, Total bilirubin, White Blood Cell Count, Magnesium
Vital signs	Diastolic Blood Pressure, PaCO2, Systolic Blood Pressure, PaO2,Mean Blood Pressure, FiO2, PaOFiO2 ratio, Respiratory Rate, Temperature (Celsius), SaO2, Heart Rate, SpO2, Arterial_BE
Output event	Total Fluid Output

5.2 Models for Comparison

- **SeqGAN** [10]**:** SeqGAN considers the sequence generation procedure as a sequential decision making process, where the generator represents a reinforcement learning agent and the discriminator indicates an evaluator to guide the generator. We replace the last layer of the generator to produce continuous medical data.
- **C-RNN-GAN** [21]**:** A method employs GANs for generating sequential continuous music. It utilizes an LSTM to represent the generator and a bidirectional LSTM as the discriminator. The discriminator performs classification for each sequence.
- **RCGAN** [5]**:** It has a similar architecture as C-RNN-GAN, except that the outputs of the generator are not fed back into the inputs and the discriminator conducts a discrimination at each time step of the sequence.
- **Imitation (RNN)**: A RNN based model which has the same structure as the generators of SC-GAN. It is also used as the pre-training process of SC-GAN.
- **SC-GAN:** Proposed model, which generates the state and medication dosage of the patients simultaneously. To reflect the clinical facts, SC-GAN uses coupled generators to produce state and medication dosage respectively, where two generators are interacted with each other.
- **SC-GAN (one G):** A variant of SC-GAN, the state of patients and the corresponding medication dosage are produced using one single generator without interaction.
- **SC-GAN** ($\lambda_{fm} = 0$)**:** A variant of SC-GAN, we set $\lambda_{fm} = 0$ and $\lambda_{adv} = 1$ in Eq. 10.
- **SC-GAN** ($\lambda_{adv} = 0$)**:** A variant of SC-GAN, we set $\lambda_{fm} = 1$ and $\lambda_{adv} = 0$.

Although SeqGAN and C-RNN-GAN are not designed to generate medical data, they are employed for generating sequential data which resembles our data type and thus can be easily adapted to the medical domain.

5.3 Quantitative Evaluation for Synthetic Data

It is challenging to evaluate the performance of GANs. Human judgment can be a candidate choice, but it is impractical and costly especially for medical records. We conduct both quantitative and qualitative analysis to evaluate the generated data, such as dimension-wise probability [12], treatment recommendation task evaluation, and Pearson correlations of real data and synthetic data [4], etc.

Dimension-Wise Probability. This metric is used to measure how the distribution of generated patient data matches the real data distribution. To be specific, we discretize the values of the generated/real patient state features and dosage with the window size of 0.1. Thus each feature or dosage has eleven value slots (i.e., 0, 0.1, ..., 1). Then we calculate the probabilities of the occurrences for each features in different slots. Take the i-th feature and m-th slot as an example. Suppose the corresponding probability of real training data denoted as $p_{i,m}$ and the value from synthetic or real test data denoted as $\hat{p}_{i,m}$. We regard $(p_{i,m}, \hat{p}_{i,m})$ as a point with x-coordinate $x = p_{i,m}$ and y-coordinate $y = \hat{p}_{i,m}$. We collect all points regarding different datasets and plot them in Fig. 3. This probability reflects the distribution of each value of the features and is a very important statistical indicator. Intuitively, if more points appear near the line $x = y$, it means the feature value distribution of $\hat{p}_{i,m}$ better matches the feature value distribution of $p_{i,m}$.

From the results, we can see SeqGAN has poor performance, where the distribution of the generated values is significantly different from that of the real data. C-RNN-GAN and RCGAN generate a set of values with high/low probability while the corresponding probability of the true data is reverse. The dimension-wise probability performance increases as we consider the dependence between medication dosage and state (compared to C-RNN-GAN and RCGAN). SC-GAN also shows better performance than Imitation, which demonstrates that the adversarial learning mechanism (SC-GAN) could work better than maximum likelihood mechanism (Imitation) in generating sequential clinical data.

Treatment Recommendation Task Evaluation. It is an important issue to utilize large amount of medical records (e.g., EHRs) to improve the quality of medical treatment especially to design proper dosage for patients [13,26]. In this section, we conduct a treatment recommendation model which designs proper medication dosage for patients to evaluate whether the synthetic data could be used for real applications, inspired by [32] which has three layers with size 58, 58 and 5 and the activation function Relu. The input is patient state and the output is the medication dosage class recommended for patients. It is trained on synthetic data and tested on real data. Two distinct generated patient-centric data: sepsis and diabetes datasets are utilized for the task. Following the previous work, we discretized the dosage of medications into 5 medication space. Thus, the number of treatment/dosage classes is five.

Table 3 shows the precision and AUROC of the recommendation results on different synthetic data. *True data* indicates that the recommendation model is

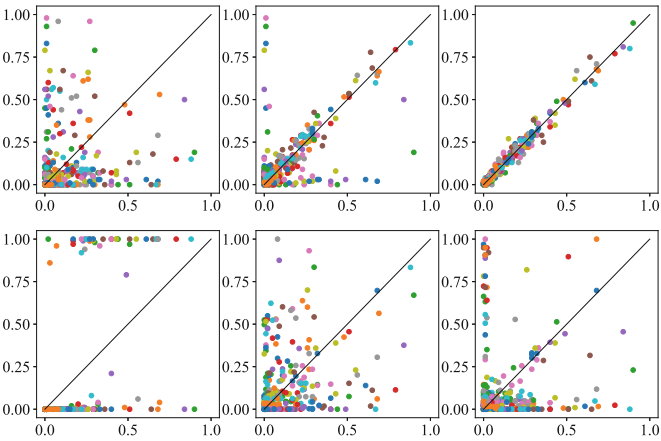


Fig. 3. Dimension-wise probability results. The top three figures, from left to right, represent the performance of Imitation, SC-GAN and True data while the below are SeqGAN, RCGAN and C-RNN-GAN. Different color corresponds to a feature of the patient. The x-axis indicates the probability for the real training data, and y-axis represents the probability for the synthetic test data or the real test data. The diagonal line is the optimal performance where the training data and test data exactly match. (Color figure online)

trained with true data and test on the true data. We randomly select 11,018 sepsis samples and 4,430 diabetes samples from generated data with same amount as real data in the training step. SeqGAN shows poor performance because it is designed for generating discrete data which may not be suitable for continuous data. RCGAN also performs not very well due to the reason that the outputs of its generator are not utilized as input for modeling. Imitation (RNN) and C-rnn-gan perform better than RCGAN, showing the benefits of leveraging outputs from last time step as current input and using RNN for modeling sequence data. SC-GAN outperforms the other baselines. The reason is that it considers the mutual interactions between drug dosage and state of patients, which reflects the clinical practice (compared with C-rnn-gan, RCGAN and SC-GAN (one)). It also utilizes the adversarial training mechanism to generate more realistic data and mitigate the exposure bias of maximum likelihood methods (compared with Imitation). By integrating both the standard loss and feature matching loss, SC-GAN achieves better performance (compared with SC-GAN ($\lambda_{adv} = 0$) and SC-GAN ($\lambda_{fm} = 0$)).

Data Augmentation Test. In addition to only using synthetic data, we train the treatment recommendation model on augmented dataset, where the synthetic data and true data are combined with the ratio of $2:3$. This experiment is utilized to show whether the generated data can bring more information and help alleviate the clinical data scarcity issue.

Table 3. Precision and AUROC of treatment recommendation task while trained on real data or synthetic data and testing on real data (%).

Methods	Sepsis-3				Diabetes		Mycosis		Tuberculosis	
	IV fluids		Vasopressor		Insulin		Fluconazole		Isoniazid	
	Pre.	AUROC	Pre.	AUROC	Pre.	AUROC	Pre.	AUROC	Pre.	AUROC
True data	**36.2**	**60.1**	**34.6**	**58.7**	**62.0**	**76.2**	**36.2**	**60.1**	**34.6**	**58.7**
C-rnn-gan	26.8	53.3	24.1	52.6	53.4	71.3	24.4	51.8	23.1	51.2
SeqGAN	18.7	50.3	18.4	50.1	41.5	66.8	17.7	50.5	17.3	50.1
Imitation (RNN)	27.1	53.7	25.3	52.8	54.1	71.8	25.3	52.5	24.0	51.4
RCGAN	25.6	52.6	23.5	51.3	51.9	70.3	24.1	51.6	22.8	50.9
SC-GAN	**31.4**	**57.1**	**29.3**	**55.9**	**56.3**	**73.1**	**30.3**	**56.2**	**27.3**	**53.6**
SC-GAN (one G)	29.9	56.2	27.9	54.1	52.9	72.2	28.2	55.1	24.7	52.3
SC-GAN ($\lambda_{adv} = 0$)	29.3	56.1	27.3	53.6	52.8	72.0	28.0	54.9	24.3	52.1
SC-GAN ($\lambda_{fm} = 0$)	28.8	55.9	27.0	53.1	52.5	71.8	27.8	54.7	24.2	52.1

As shown in Table 4, the data generated by SeqGAN and RCGAN seems to make no contribution to the real data because results show no improvements over those of only employing real data, showing no additional useful information is generated. Imitation and C-rnn-gan behave better for their contribution to improving recommendation performance. Finally, by adding the data generated by SC-GAN, the recommendation model achieves significantly better results, demonstrating the generated data indeed provide additional useful information to complement true data and it is good for data augmentation.

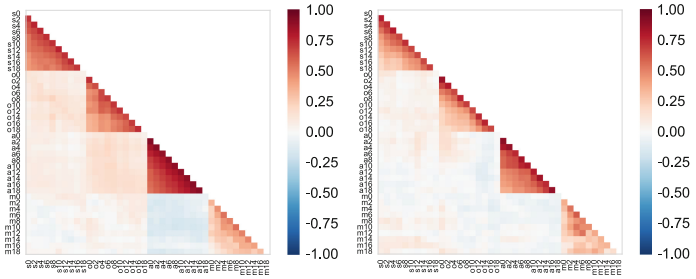
Table 4. Precision and AUROC of medication dosage recommendation task trained with both synthetic and real training data and tested on real data (%).

Methods	Sepsis-3				Diabetes		Mycosis		Tuberculosis	
	IV fluids		Vasopressor		Insulin		Fluconazole		Isoniazid	
	Pre.	AUROC	Pre.	AUROC	Pre.	AUROC	Pre.	AUROC	Pre.	AUROC
True data	36.2	60.1	34.6	58.7	62.0	76.2	48.5	69.2	47.3	68.7
C-rnn-gan	36.9	60.4	33.4	58.1	62.9	76.6	36.6	63.7	32.4	61.3
SeqGAN	32.2	58.3	30.0	57.2	54.3	73.2	25.3	57.7	24.8	52.0
Imitation (RNN)	37.3	60.7	34.8	**59.2**	63.2	76.9	36.5	64.8	33.6	61.6
RCGAN	35.4	59.7	33.2	58.0	61.4	76.3	36.1	63.4	32.3	61.1
SC-GAN	**38.6**	**61.4**	**35.7**	**59.2**	**64.6**	**77.3**	**39.2**	**65.1**	**34.4**	**62.5**
SC-GAN (one G)	37.1	60.7	33.8	59.3	62.7	76.4	37.3	62.5	32.1	61.3
SC-GAN ($\lambda_{adv} = 0$)	36.5	59.8	33.1	58.6	61.8	76.0	37.1	62.3	31.8	61.2
SC-GAN ($\lambda_{fm} = 0$)	36.2	59.3	32.9	58.0	61.4	75.7	36.8	62.1	31.5	61.0

5.4 Qualitative Evaluation for Synthetic Data

Pairwise Pearson Correlation. Following [4], we adopt Pearson correlation coefficient (PCC) [33] to obtain Pearson correlation structures of feature pairs for real data and synthetic data, respectively. The value of PCC ranges from +1 to −1, where 1 represents complete positive correlation, and −1 corresponds to complete negative correlation. We select three features of the patients (systolic blood pressure, spo2 and Arterial_BE) and the dosage of Intravenous Fluids to conduct Pearson correlation experiments. The Pearson correlation structures of the real data is in Fig. 4(a) and generated data is in Fig. 4(b).

As shown in Fig. 4, s, o, a, m represent systolic blood pressure, spO2, Arterial_BE and Intravenous Fluids respectively. The numbers 0–18 indicate the ICU stay length (hour) of the sepsis patients. Here we only extract ten time steps for comparison due to the space limitation. Both the synthetic data and real data show the Intravenous Fluids has positive correlation with systolic blood pressure and spo2. But for Arterial_BE and Intravenous Fluids, the synthetic data shows weaker correlation than the real data. The synthetic data generated by SC-GAN shows a little different result compared to the true data. The main trends of the results remain consistent.



(a) Real Data (b) Synthetic Data

Fig. 4. Pairwise Pearson correlation (PPC) between time series features.

Generated Patient-Centric Data. To conduct a qualitative evaluation for synthetic data, we randomly select eight features generated by SC-GAN to intuitively compare the difference between them and the true data. We also invite two clinicians to evaluate the results.

For most of the generated data, the clinicians could not judge they are synthetic, except for paO2. This is because paO2 involves frequent variation. As shown in Fig. 5, the trends of the generated features are not regular. Because the state of these patients can change significantly as the time goes on. However, the mean of these data could be concentrated. Figure 5 shows that the values of different patient features vary in different regions. The synthetic data produced by SC-GAN obtains the similar values regions as the true data.

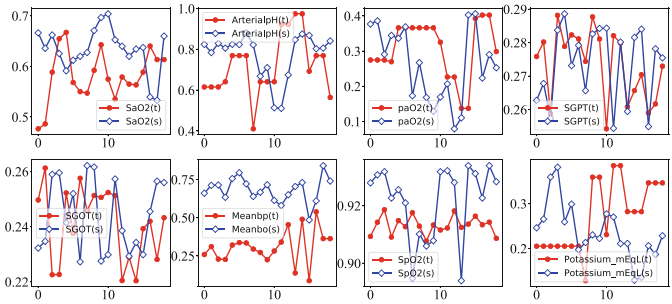


Fig. 5. Generated trajectory data lasting from initial visit to 19 * 4 h for specified features ("t" indicates the true data and "s" indicates the synthetic data)

6 Conclusion

In this paper, we have proposed SC-GAN to generate sequential and continuous medical data including the state of patients and the dosage of medications the patients take. The main novelty of the model is the coupled generators which coordinate the generation of patient state and medication dosage to capture the mutual interactions between medications and patient state. The comprehensive experiments on the real world EHRs dataset demonstrate the data generated by SC-GAN can not only gain performance close to the real data on treatment recommendation task, but also be useful for data augmentation.

Acknowledgements. This work is partially supported by the National Key Research and Development Program of China under Grant No. 2016YFB1000904, NSFC (61702190, U1609220).

References

1. Gostin, L.O., et al.: Beyond the HIPAA Privacy Rule: Enhancing Privacy. Improving Health Through Research. National Academies Press, Washington, DC (2009)
2. McLachlan, S., Dube, K., Gallagher, T.: Using the caremap with health incidents statistics for generating the realistic synthetic electronic healthcare record. In: Healthcare Informatics (ICHI), pp. 439–448 (2016)
3. Buczak, A.L., Babin, S., Moniz, L.: Data-driven approach for creating synthetic electronic medical records. BMC Med. Inform. Decis. Making **10**, 59 (2010)
4. Beaulieu-Jones, B.K., et al.: Privacy-preserving generative deep neural networks support clinical data sharing, p. 159756. BioRxiv, C.S. (2017)
5. Yahi, A., Vanguri, R., Elhadad, N., Tatonetti, N.P.: Generative adversarial networks for electronic health records: a framework for exploring and evaluating methods for predicting drug-induced laboratory test trajectories. In: NIPS (2017)
6. Goodfellow, I., et al.: Generative adversarial nets. In: NIPS, pp. 2672–2680 (2014)
7. Isola, P., Zhu, J., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR, pp. 5967–5976 (2017)

8. Gregor, K., Danihelka, I., Graves, A., Rezende, D.J., Wierstra, D.: DRAW: a recurrent neural network for image generation. In: ICML, pp. 1462–1471 (2015)
9. Odena, A., Olah, C., Shlens, J.: Conditional image synthesis with auxiliary classifier GANs. In: ICML, pp. 2642–2651 (2017)
10. Yu, L., Zhang, W., Wang, J., Yu, Y.: SeqGAN: sequence generative adversarial nets with policy gradient. In: AAAI, pp. 2852–2858 (2017)
11. William, F., Goodfellow, I., Dai, A.M.: MaskGAN: better text generation via filling in the _. In: ICLR (2018)
12. Choi, E., Biswal, S., Malin, B., Duke, J., Stewart, W.F., Sun, J.: Generating multi-label discrete electronic health records using generative adversarial networks. Machine Learning for Healthcare (2017)
13. Raghu, A., Komorowski, M., Celi, L.A., Szolovits, P., Ghassemi, M.: Continuous state-space models for optimal sepsis treatment: a deep reinforcement learning approach. In: Proceedings of the Machine Learning for Health Care, pp. 147–163 (2017)
14. Wang, L., Zhang, W., He, X., Zha, H.: Supervised reinforcement learning with recurrent neural network for dynamic treatment recommendation. In: KDD, pp. 2447–2456. ACM (2018)
15. Waechter, J., et al.: Interaction between fluids and vasoactive agents on mortality in septic shock: a multicenter, observational study. Criti. Care Med. **42**(10), 2158–2168 (2014)
16. Denton, E.L., et al.: Deep generative image models using a Laplacian pyramid of adversarial networks. In: NIPS, pp. 1486–1494 (2015)
17. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: NIPS, pp. 3104–3112 (2014)
18. Casella, P., Paiva, A.: MAgentA: an architecture for real time automatic composition of background music. In: de Antonio, A., Aylett, R., Ballin, D. (eds.) IVA 2001. LNCS (LNAI), vol. 2190, pp. 224–232. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44812-8_18
19. Zhu, H., et al.: Xiaoice band: a melody and arrangement generation framework for pop music. In: KDD, pp. 2837–2846 (2018)
20. Bengio, S., Vinyals, O., Jaitly, N., Shazeer, N.: Scheduled sampling for sequence prediction with recurrent neural networks. In: NIPS, pp. 1171–1179 (2015)
21. Mogren, O.: C-RNN-GAN: continuous recurrent neural networks with adversarial training. CoRR abs/1611.09904 (2016)
22. Office for Civil Rights: Guidance regarding methods for de-identification of protected health information in accordance with the health insurance portability and accountability act (HIPAA) privacy rule. U.S. Department of Health and Human Services (2013)
23. Li, C.Y., Liang, X., Hu, Z., Xing, E.P.: Hybrid retrieval-generation reinforced agent for medical image report generation. arXiv preprint arXiv:1805.08298 (2018)
24. Esteban, C., Hyland, S.L., Rätsch, G.: Real-valued (medical) time series generation with recurrent conditional GANs. arXiv preprint arXiv:1706.02633 (2017)
25. Hoang, Q., Nguyen, T.D., Le, T., Phung, D.: Multi-generator generative adversarial nets. In: ICLR (2017)
26. Wang, L., Zhang, W., He, X., Zha, H.: Personalized prescription for comorbidity. In: Pei, J., Manolopoulos, Y., Sadiq, S., Li, J. (eds.) DASFAA 2018. LNCS, vol. 10828, pp. 3–19. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91458-9_1
27. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training GANs. In: NIPS, pp. 2234–2242 (2016)

28. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
29. Johnson, A.E., et al.: MIMIC-III, a freely accessible critical care database. Sci. Data **3**, 160035 (2016)
30. Weng, W.H., Gao, M., He, Z., Yan, S., Szolovits, P.: Representation and reinforcement learning for personalized glycemic control in septic patients. In: NIPS Workshop (2017)
31. Singer, M., et al.: The third international consensus definitions for sepsis and septic shock (sepsis-3). JAMA **315**(8), 801–810 (2016)
32. Bajor, J.M., ALasko, T.: Predicting medications from diagnostic codes with recurrent neural networks. In: ICLR (2017)
33. Pearson, K.: Notes on regression and inheritance in the case of two parents. Proc. R. Soc. London **58**, 240–242 (1895)