GluGAN: Generating Personalized Glucose Time Series Using Generative Adversarial Networks

Article *in* IEEE Journal of Biomedical and Health Informatics · May 2023

DOI: 10.1109/JBHI.2023.3271615

4 authors, including:

Taiyu Zhu
University of Oxford
25 PUBLICATIONS **873** CITATIONS
SEE PROFILE

Pau Herrero
Imperial College London
155 PUBLICATIONS **3,067** CITATIONS
SEE PROFILE

GluGAN: Generating Personalized Glucose Time Series Using Generative Adversarial Networks

Taiyu Zhu, *Graduate Student Member, IEEE*, Kezhi Li, *Member, IEEE*, Pau Herrero, *Member, IEEE*, and Pantelis Georgiou, *Senior Member, IEEE*

***Abstract*— Time series data generated by continuous glucose monitoring sensors offer unparalleled opportunities for developing data-driven approaches, especially deep learning-based models, in diabetes management. Although these approaches have achieved state-of-the-art performance in various fields such as glucose prediction in type 1 diabetes (T1D), challenges remain in the acquisition of large-scale individual data for personalized modeling due to the elevated cost of clinical trials and data privacy regulations. In this work, we introduce GluGAN, a framework specifically designed for generating personalized glucose time series based on generative adversarial networks (GANs). Employing recurrent neural network (RNN) modules, the proposed framework uses a combination of unsupervised and supervised training to learn temporal dynamics in latent spaces. Aiming to assess the quality of synthetic data, we apply clinical metrics, distance scores, and discriminative and predictive scores computed by post-hoc RNNs in evaluation. Across three clinical datasets with 47 T1D subjects (including one publicly available and two proprietary datasets), GluGAN achieved better performance for all the considered metrics when compared with four baseline GAN models. The performance of data augmentation is evaluated by three machine learning-based glucose predictors. Using the training sets augmented by GluGAN significantly reduced the root mean square error for the predictors over 30 and 60-minute horizons. The results suggest that GluGAN is an effective method in generating high-quality synthetic glucose time series and has the potential to be used for evaluating the effectiveness of automated insulin delivery algorithms and as a digital twin to substitute for pre-clinical trials.**

***Index Terms*— Artificial intelligence (AI), continuous glucose monitoring (CGM), diabetes, generative adversarial network (GAN), glucose time series.**

I. Introduction

DIABETES is a group of metabolic disorders characterized by hyperglycemia, which affects almost a half billion people in the world [1]. Due to the destruction of pancreatic β cells, people living with type 1 diabetes (T1D) suffer from absolute insulin deficiency and require lifelong management to maintain blood glucose concentration in a therapeutic range (e.g. [70, 180] mg/dL). Otherwise, adverse glycemic events, including hyperglycemia and hypoglycemia, would lead to short- and long-term diabetes-related complications, such as retinopathy, neuropathy, cerebrovascular disease, and coronary heart disease [2]. T1D management involves consistent adherence to treatment regimens on a daily basis, such as exogenous insulin administration and dietary planning. Furthermore, monitoring glucose levels is a cornerstone of diabetes management. Glucose data are key indicators in glycaemic control by allowing to adjust existing therapy [3], [4] and enable healthcare providers to perform glycemic analysis for further modification of individual treatment [5]. In general, there are two common glucose monitoring systems: self-monitoring blood glucose (SMBG) and continuous glucose monitoring (CGM). SMBG is the most conventional and widely used method, which requires finger-pricking to obtain capillary blood samples and a glucose meter to analyze and report the results [6]. People using SMBG tend to finger-prick three to four times per day, but this is usually not enough to present a comprehensive profile of glucose trajectories and effectively prevent undesired glycemic events [7], [8]. To address this challenge, CGM technology has been developed over the past two decades [9]–[11]. Using subcutaneously inserted sensors under the skin, CGM can measure glucose levels in the interstitial fluid and estimate plasma glucose with a fixed time period, e.g., every five minutes. CGM might require periodic calibrations based on SMBG measurements, and the most recent models come with factory calibration [12].

Fig. 1 shows clinical settings of T1D management with glucose monitoring and insulin administration. Taking advantage of widespread mHealth, Internet of things, and edge computing technologies [13]–[15], many people with T1D regularly record daily events, such as food carbohydrates, meal insulin bolus, and SMBG measurements, via smartphone-based apps. The vast amount of glucose data generated by CGM has allowed the application of artificial intelligence technologies to improve T1D management [16], [17]. Technological progress in the field of deep learning has offered the promise of developing new state-of-the-art personalized algorithms in the fields of automatic insulin delivery [18]–[21] and glucose pre-

This work was supported by EPSRC EP/P00993X/1 and President's Ph.D. Scholarship at Imperial College London. *(Corresponding author: K. Li)*

T. Zhu, P. Herrero, P. Georgiou are with Centre for Bio-inspired Technology, Imperial College London, London, United Kingdom. (e-mail: {taiyu.zhu17, pherrero, pantelis}@imperial.ac.uk).

K. Li is with Institute of Health Informatics, University College London, London, United Kingdom. (e-mail: ken.li@ucl.ac.uk).

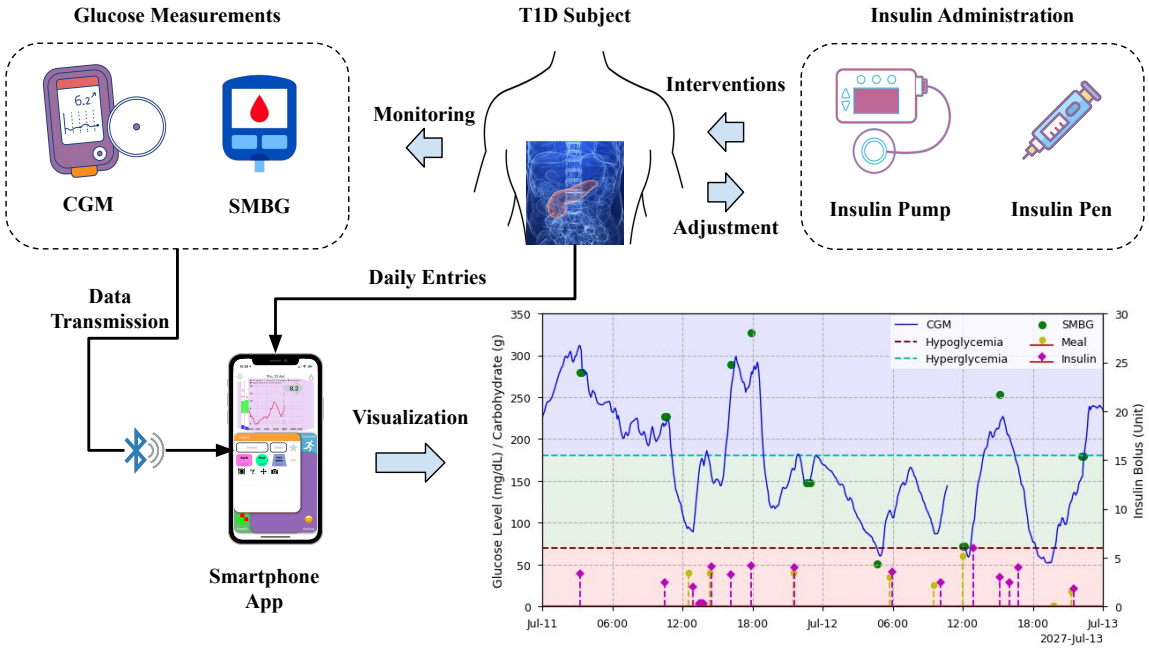

Fig. 1: Illustration of a T1D management system. A smartphone app is used to communicate with the glucose monitoring devices via Bluetooth connectivity, collect daily logs, and visualize historical profiles and current trends of glucose levels. The bottom right plot shows two-day multivariate time series data of a clinical T1D subject in the OhioT1DM dataset.

diction [22]. In particular, a wide range of architectures of deep neural networks, such as convolutional neural networks [23], [24] and recurrent neural networks (RNNs) [25]–[27], have been applied to glucose prediction with CGM data for short- and medium-term prediction horizons, e.g., 30 and 60 minutes.

However, in diabetes management, applying deep learning or other data-driven algorithms is often challenging. It is because training these models usually requires collecting large-scale datasets in months of clinical trials. This problem is referred to as the cold start issue [22]. In this regard, glucose-insulin simulators, such as the FDA-accepted UVA/Padova T1D simulator [28], have been developed to enable cost-effective *in silico* trials. These simulators can be used to generate synthetic glucose time series using predefined virtual cohorts. Nevertheless, exploiting outpatient data of T1D subjects under free-living conditions for personalized data generation [29] is still an open problem. To this end, generative adversarial network (GAN)-based frameworks can provide effective solutions. As a group of deep generative models, GANs consist of at least two deep neural networks acting as the generator and discriminator, which are trained in an adversarial process [30]. GANs have been widely used to generate synthetic image data to improve deep learning models for computer vision tasks [31]. In recent works, GAN-based frameworks for sequential data generation have attracted significant attention. In [32], a GAN model was first applied to generate music using bidirectional RNNs with long short-term memory (LSTM) cells. Esteban *et al.* [33] extended LSTM-based GANs to the recurrent conditional GAN (RCGAN), which allow generating real-valued medical time series data with conditional inputs and a differentially private training procedure. Based on dilated convolutional neural networks, WaveGAN was proposed to produce synthetic audio [34]. Yoon *et al.* [35] proposed TimeGAN by introducing an embedding network and a recovery network in the GAN architecture to learn hidden temporal dynamics. The performance was tested in multiple time series datasets, including sinusoidal sequences, stock prices, energy data, and discrete events. In a more recent study, TimeGAN was applied to generate synthetic hypoglycemic events to tackle the issue of imbalanced data in glycemic classification [36]. Previously, we also explored a modified GAN-based model to extract feature maps from a multivariate input and forecast glucose levels [37].

In this article, we propose GluGAN, which to the best of our knowledge is the first GAN framework that allows generating realistic glucose time series, presenting a first step towards data-driven personalized T1D simulators. Different from existing simulators based on fixed glucose profiles, GluGAN can learn patterns from glucose readings for T1D individuals and generate personalized glucose data based on outpatient T1D datasets. GluGAN allows manually recorded daily entries in T1D management as conditional inputs (e.g. SMBG measurements, carbohydrates from meal intake, and insulin delivery), and uses a combination of losses to learn temporal patterns of the time series data. In this work, the proposed use cases for GluGAN include the generation of ambulatory glucose profiles and data augmentation to improve the performance of machine learning-based short- and mid-term glucose prediction algorithms. The framework can be extended and transferred in developing personalized simulators and digital twins [38], [39].

II. METHODS

A. Problem Formulation

Given personalized data from a clinical T1D dataset, we first slice the multivariate time series by a sliding window and obtain input sequences of length L. We assume an input sequence starts from a timestep ts and ends with a timestep t, where $ts = t + 1 - L$.

As shown in Fig. 1, meal ingestion and insulin delivery can cause significant glucose fluctuations. Furthermore, CGM is usually calibrated with SMBG measurements when sensor replacement occurs. Discrepancies between CGM and SMBG are usually observed, especially in the postprandial period, which is a common phenomenon called meal-related glucose differences [40]. Meal, insulin, and SMBG data are therefore highly correlated to glucose dynamics and can provide rich information for generating realistic glucose time series. There are strong temporal dependencies between glucose levels and these exogenous features, and it is difficult to map them as the static features proposed in [35], due to the large variability in real-world scenarios. Therefore, the conditional inputs of meal carbohydrates, insulin bolus, and SMBG data, $\mathbf{C}_{ts:t} \in \mathcal{R}^{3\times L}$, used in this work are multivariate time series with a length L and the same resolution of CGM, hence their values are equal to zero most of the time. We attach the conditional inputs to the CGM time series $\mathbf{G}_{ts:t} \in \mathcal{R}^{1\times L}$ and have the real input data of embedding network $\mathbf{x}_{ts:t} = [\mathbf{G}_{ts:t}; \mathbf{C}_{ts:t}] \in \mathcal{R}^{4\times L}$, as shown in the bottom right plot of Fig. 1. Similarly, the input of the synthetic data generator is the concatenation of a random vector $\mathbf{Z}^{sp} \in \mathcal{R}^{1\times L}$ sampled from the Wiener process and the same conditional inputs, which is denoted by $\mathbf{z}_{ts:t} = [\mathbf{Z}^{sp}_{1:L}; \mathbf{C}_{ts:t}] \in \mathcal{R}^{4\times L}$. In this article, the conditional inputs $\mathbf{C}_{ts:t}$ are based on real data, so the size of synthetic data is the same as that of real data. The work can be extended to generate unlimited synthetic data when the conditional inputs are simulated given their statistical characteristics.

In this context, GluGAN aims at learning a density similar to the distribution of ground-truth data, when conditioned on the same auxiliary information. This is the objective of a conditional GAN framework, which can be defined as follows [30]:

$$\min_{\hat{p}} \mathrm{JS}(p(\mathbf{G}_{ts:t}|\mathbf{C}_{ts:t})||\hat{p}(\mathbf{G}_{ts:t}|\mathbf{C}_{ts:t})) \tag{1}$$

where JS is the Jensen–Shannon (JS) divergence to measure of similarity between two probability distributions; p stands for the density function of the distribution over real data, while $\hat{p}$ is an approximate distribution of the generator's outputs. JS divergence is a symmetric version of conventional Kullback–Leibler (KL) divergence, which is recommended to derive the adversarial loss of GANs [41]. However, the JS divergence requires an optimal value for discriminator, i.e, perfect adversary, which is difficult to be obtained in unsupervised learning.

Fortunately, temporal relationships of time series data can guide the generation of sequential data, especially for the time series with high correlation across timesteps. Therefore, we introduce another objective function that focuses on the step-wise conditional distributions, which can be formulated as [35]:

$$\min_{\hat{p}} \mathrm{KL}(p(G_t|\mathbf{G}_{ts:t-1}, \mathbf{C}_{ts:t-1})||\hat{p}(G_t|\mathbf{G}_{ts:t-1}, \mathbf{C}_{ts:t-1})) \tag{2}$$

where KL denotes the KL divergence. Although this divergence is an asymmetric measure, it can be optimized by supervised learning with maximum likelihood estimation [35]. Therefore, a supervised loss is employed in the adversarial training to learn the transition dynamics.

B. GluGAN Architecture

To generate realistic personalized glucose time series, we develop GluGAN by modifying a standard GAN architecture in two ways. First, in addition to the generator discriminator, we introduce three other modules into GluGAN, including an embedding network, a recovery network, and a supervisor network. The embedding and recovery networks are used for auto-encoding to project time series data into a lower-dimensional latent space, where both adversarial learning and supervised learning are performed, aiming to improve generation performance for high-dimensional time series [35]. The supervisor network is used to learn the step-wise dynamics with the targets derived from real sequences. These three additional modules are jointly trained with the generator and discriminator. Each module consists of a four-layer RNN with gated recurrent units, of which the hyperparameters are determined in model validation. By applying maximum likelihood estimation, we use a mean square error loss function for supervised learning to optimize the objective in Equation (2) and combine it with the unsupervised adversarial loss to optimize the objective in Equation (1). It should be noted that, in the loss functions of model training, the target $\mathbf{G}$ and conditional inputs $\mathbf{C}$ in these two equations are replaced by the latent representations, due to the use of embedding space. In this case, the model not only learns to generate data with similar distribution but also to capture the temporal dynamics of time series. Secondly, we employ three conditional input features [33] to indicate the underlying glycemic states, including carbohydrate amount of meal ingestion, bolus insulin, and SMBG measurements.

Fig. 2 depicts the overall architecture of the proposed GluGAN. For auto-encoding purposes, we employ the embedding network to convert the real input features to latent representations $\mathbf{h}$ and reconstruct glucose data $\widetilde{G}_{ts:t}$ through the recovery network $\mathcal{R}$. Similarly, given a random input vector, the generator $\mathcal{G}$ outputs the synthetic embedding vector $\hat{\mathbf{e}}$, which is disciplined by the supervisor $\mathcal{S}$ to learn step-wise temporal dynamics for the synthetic latent vector $\hat{\mathbf{h}}$. Then, we obtain the final output of synthetic glucose time series $\hat{G}_{ts:t}$ by the mapping function of the recovery network. Instead of directly comparing the outcomes of the generator with real data, the discriminator $\mathcal{D}$ of GluGAN performs classification in the latent space. We denote the outputs of the discriminator by $y_t, \hat{y}_t \in \{0, 1\}$ for the real and synthetic model input data, respectively.

Correspondingly, a total of three losses is used to optimize the weights of GluGAN. To obtain reliable conversion between

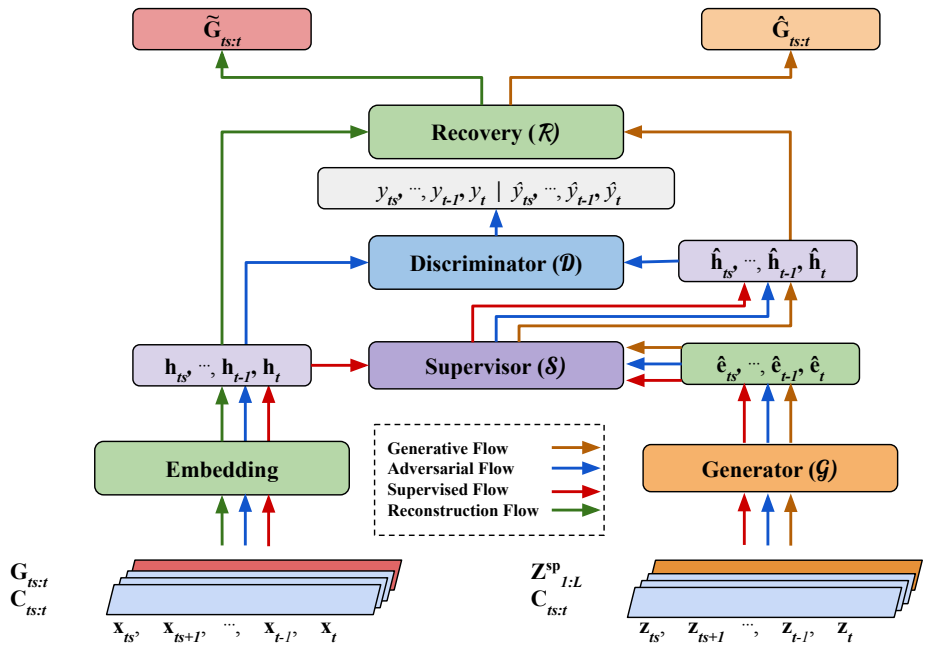

Fig. 2: System architecture of the proposed GluGAN. The data flows corresponding to glucose time series generation, adversarial training, supervised learning, and reconstruction are marked with orange, blue, red, and green arrows, respectively.

latent space and glucose features, a reconstruction loss $\mathcal{L}_R$ is applied to train embedding and recovery modules, which is given by:

$$\mathcal{L}_R = \mathbb{E}_{\mathbf{x}\sim p}[\sum_t \left\| \widetilde{G}_t - G_t \right\|_2]. \tag{3}$$

Meanwhile, supervised learning with a loss $\mathcal{L}_S$ is performed to minimize the step-wise differences between synthetic and real latent vectors, when the current synthetic embedding vector $\hat{\mathbf{e}}_t$ are conditioned on the real latent sequences at previous timesteps ($\mathbf{h}_{ts:t-1}$). It is formulated as:

$$\mathcal{L}_S = \mathbb{E}_{\mathbf{x}\sim p, \mathbf{z}\sim \hat{p}} \sum_t \|\mathbf{h}_t - \mathcal{S}(\hat{\mathbf{e}}_t, \mathbf{h}_{ts:t-1})\|_2 \,, \tag{4}$$

where $\mathcal{S}$ represents the function of the supervisor network. Similar to the standard GAN framework, the discriminator is trained to be better at discriminating real from synthetic data, while the generator is designed to generate sequences that are indistinguishable from real glucose time series. Hence, we treat the optimization as a two-player minimax game. The unsupervised losses of adversarial training are calculated using the classification results $y_t, \hat{y}_t$, which are given by:

$$\begin{aligned} \mathcal{L}_U^{\mathcal{G}} &= \mathbb{E}_{\mathbf{z}\sim\hat{p}} \sum_t \log(1-\hat{y}_t), \\ \mathcal{L}_U^{\mathcal{D}} &= -\mathbb{E}_{\mathbf{x}\sim p} \sum_t \log y_t - \mathbb{E}_{\mathbf{z}\sim\hat{p}} \sum_t \log(1-\hat{y}_t), \end{aligned} \tag{5}$$

where $\mathcal{L}_U^{\mathcal{G}}$ and $\mathcal{L}_U^{\mathcal{D}}$ denote the unsupervised losses of the generator and discriminator, respectively.

C. Model Development

Data preproccessing is an essential step to obtain high-quality multivariate time series from clinical datasets. We first remove outliers for each input feature with a set of maximum and minimum thresholds based on physiological features. In particular, we exclude negative values for each feature, insulin doses above 50 units, and glucose values above 500 mg/dL. Notably, there are many missing gaps in CGM measurements (Fig. 1), due to various reasons, such as sensor replacement and calibration, sensor noise, and signal loss. Thus, we perform linear interpolation to fill the gaps in the middle of CGM sequences and use linear extrapolation for missing data samples on the tails [24], [25], [37]. We also exclude the CGM sequences with a gap longer than 15 minutes. All the input features are normalized to a range of [0,1] with the min-max normalization.

Algorithm 1 presents the details of the model development. Training the GluGAN model (Fig. 2) requires optimizing all the network modules with the loss functions defined in Equations (3), (4), and (5). Given the numbers $\mathcal{T}_R, \mathcal{T}_S, \mathcal{T}_J$ of training iterations for embedding learning, supervised learning, and joint learning, respectively, we optimize the embedding and recovery networks with ground truth data and $\mathcal{L}_R$, and then train the supervisor module alone with $\mathcal{L}_S$. Finally, all five modules are jointly trained with the combinations of unsupervised and supervised losses. Specifically, two hyperparameters λ_1, λ_2 are employed to adjust the ratios of $\mathcal{L}_S$ when combined with reconstruction and unsupervised losses. To achieve a Nash equilibrium for the two-player non-cooperative game and avoid the discriminator becoming too strong in adversarial training [42], we update the generator more frequently with an inner loop and update the discriminator only if $\mathcal{L}_U^{\mathcal{D}}$ is above a predefined threshold l_D [32]. During the testing phase, GluGAN can generate synthetic glucose data with the batch inputs of testing data. Table II in the Appendix summarizes

Algorithm 1 Developing GluGAN to Generate Glucose Time Series

Input: a preprocessed training set $\mathcal{D}_{tr}$ and synthetic inputs $\mathcal{P}_Z$; iteration numbers $\mathcal{T}_R, \mathcal{T}_S, \mathcal{T}_J$; loss ratios λ_1, λ_2; iterations of the inner loop k, threshold of discriminator loss l_D.
Model training
for iterations in $\mathcal{T}_R$ **do**
  Sample mini-batches from $\mathcal{D}_{tr}$
  Update the weights of the embedding and recovery networks by minimizing loss $\mathcal{L}_R$
end for
for iterations in $\mathcal{T}_S$ **do**
  Sample mini-batches from $\mathcal{D}_{tr}$ and from $\mathcal{P}_z$
  Update the weights of the supervisor network by minimizing $\mathcal{L}_S$
end for
for iterations in $\mathcal{T}_J$ **do**
  for iterations in k **do**
    Sample mini-batches from $\mathcal{D}_{tr}$ and $\mathcal{P}_z$
    Update the weights of the supervisor and generator by minimizing $\lambda_1\mathcal{L}_S + \mathcal{L}_U^{\mathcal{G}}$
    Update the weights of the embedding and recovery networks by minimizing $\lambda_2\mathcal{L}_S + \mathcal{L}_R$
  end for
  Sample mini-batches from $\mathcal{D}_{tr}$ and $\mathcal{P}_z$
  if $\mathcal{L}_U^{\mathcal{D}} > l_D$ **then**
    Update the weights of the discriminator by minimizing $\mathcal{L}_U^{\mathcal{D}}$
  end if
end for
Model testing
Given the batch of $\{\mathbf{z}_{ts:t}\}$ from a testing set $\mathcal{D}_{te}$
Obtain synthetic glucose time series by $\hat{\mathbf{G}}_{ts:t} = \mathcal{R}(\mathcal{S}(\mathcal{G}(\mathbf{z}_{ts:t})))$

the values of the hyperparameters used in this work.

D. Clinical Datasets

The performance of GluGAN is tested on three real datasets: the OhioT1DM [43], ARISES, and ABC4D datasets, which are collected from months of trials with different clinical settings. The OhioT1DM dataset is publicly available [43] and contains the eight weeks' data of 12 T1D subjects who used Medtronic Enlite CGM and Medtronic 530G or 630G insulin pumps for continuous subcutaneous insulin infusion (CSII) treatment. The ARISES and ABC4D datasets are proprietary datasets (Imperial College London, London, UK). In particular, the ARISES dataset contains the data from 12 T1D adults over a six-week clinical trial (NCT03643692). The study was under the protocol (18/LO/1096) approved by London - Fulham Research Ethics Committee in 2018. The participants wore Dexcom G6 CGM and were equally stratified by mode of insulin delivery, including CSII and multiple daily injections (MDIs). The ABC4D dataset includes the data of 25 T1D subjects who wore Dexcom G5 CGM and were on MDI therapy, which was collected from a six-month clinical trial (NCT02053051) [44]. The study was under the protocol (13/LO/0264) approved by London - Chelsea Research Ethics Committee in 2013. The demographic characteristics of the three datasets are shown in Table III in the Appendix.

III. Experiments

A. Data Splitting and Analysis

During model development, we split datasets into training sets $\mathcal{D}_{tr}$ and testing sets $\mathcal{D}_{te}$. The OhioT1DM dataset is provided with a training set and a testing set for each T1D subject, which respectively contain data of around 40 and 10 days [43]. As for the subjects in ARISES and ABC4D datasets, we use the first 80% data as training sets and the rest 20% data as hold-out testing sets. It is a common split method for developing machine learning algorithms in T1D-related tasks, which guarantees that future information is not involved in current model inference [22]. For the training sets in each dataset, we use the first 75% data for model training, while the last 25% data are used as hold-out validation sets to tune hyperparameters. We ensure that there is no data leakage by strictly following chronological partition to split data for each subject, which guarantees that the validation and testing sets do not include any data in the training set. We generate synthetic datasets $\hat{\mathcal{D}}_{tr}$ and $\hat{\mathcal{D}}_{te}$ for training and testing sets, respectively, aiming at following the train-on-synthetic and test-on-real (TSTR) routine [33] to test model performance. It should be noted that this data split, as shown in Fig. 7a of the Appendix, is only applied to evaluate the performance of all the considered GAN models in terms of the quality of synthetic data.

In the use case of glucose prediction, we test the performance of GluGAN and data augmentation through a transfer-learning framework [25], [36]. Assuming that only the first two weeks of glucose data are available in each training set, we combine these data as a global set to develop a population GluGAN model and fine-tune the whole model with individual training data of a hold-out subject. In particular, we perform the leave-one-out cross-validation (LOOCV) [36] in model validation, where a subject is randomly selected as the validation set and the data of the remaining subjects are used for model training. This particular length is selected because the lifespan of most commercial CGM sensors is within 14 days. Then the personalized GluGAN model generates two-week synthetic glucose data, which are combined with the original data to develop an augmented training set. We compare the performance of glucose prediction using augmented training sets (train-on-augmented and test-on-real (TATR)) and original training sets (train-on-real and test-on-real (TRTR)). Fig. 7b of the Appendix depicts the TATR and TRTR process.

In Fig. 3, we plot the autocorrelogram of glucose time series using consecutive sequences with a minimum length of three days. It is to be noted that, for the three considered datasets, the glucose data have high autocorrelation when time lags are smaller than 105 minutes. Therefore, it is important to introduce the autoregressive prior and supervised learning loss into GluGAN model to learn step-wise temporal dynamics of glucose time series.

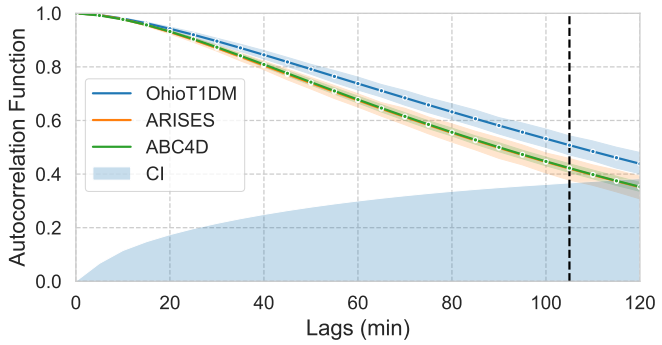

Fig. 3: Autocorrelation function (ACF) of glucose time series in the three datasets with lags up to 120 minuets. The ACF of the OhioT1DM, ARISES, and ABC4D datasets are respectively showed in blue, orange, and green solid lines with 95% bootstrap confidence intervals (CIs). The blue shaded region indicates the 95% Bartlett CI, and any ACF value outside this region is statistically different from zero. The vertical dashed line indicates the maximum time lag (105 min) with significant ACF values for the three datasets.

B. Evaluation Metrics

To comprehensively evaluate the performance of generative models, we use a set of commonly employed metrics in previous work on GAN frameworks and time series data [33], [35]. The similarity and diversity of synthetic and real testing glucose data (i.e., $\hat{\mathcal{D}}_{te}$ and $\mathcal{D}_{te}$) are qualitatively visualized by principal component analysis (PCA) [45] and t-distributed stochastic neighbor embedding (t-SNE) [46]. We convert temporal L-dimension sequential data into two dimensions and plot the distributions, where each dot stands for a glucose sequence.

Quantitative analysis is also performed using six metrics. First, by assigning 1 and 0 labels to real and synthetic data, respectively, we treat it as a binary classification problem and use a post-hoc classifier based on an RNN model with two LSTM layers [35]. It is trained on a merged training sets (i.e., $\hat{\mathcal{D}}_{tr}$ and $\mathcal{D}_{tr}$) and tested on a merged testing sets (i.e., $\hat{\mathcal{D}}_{te}$ and $\mathcal{D}_{te}$). We calculate a discriminative score as $|accuracy - 0.5|$. If the accuracy of the post-hoc RNN is close to 0.5 (i.e., random guess), the discriminative score decreases, and it indicates that synthetic data are indistinguishable from real data. Similarly, we introduce a step-wise predictive score with a post-hoc predictive model to predict next-step glucose value and calculate scores of the root mean square error (RMSE). The predictive model is also constructed by an RNN model with two LSTM layers, which is trained on synthetic datasets ($\hat{\mathcal{D}}_{tr}$) and tested on real datasets ($\mathcal{D}_{te}$), i.e., TSTR. Low predictive scores indicate the synthetic data are useful in terms of prediction tasks.

To investigate the clinical significance of the generated synthetic data, we calculate the mean absolute error (MAE) between the percentage time below range (TBR) and percentage time above range (TAR) of real data and those of synthetic data. TBR and TAR are the clinical targets recommended by the International Consensus [47]. The MAE of TBR and MAE of TAR respectively indicate whether GAN models can accurately estimate the percentages of time spent in hypoglycemia and hyperglycemia. Furthermore, the earth mover's distance (EMD) and maximum mean discrepancy (MMD) with a radial basis function kernel [33] are introduced to measure the distance between the probability distribution of real data and that of synthetic data.

In the experiments of glucose prediction with data augmentation, RMSE and MAE are used to evaluate the prediction accuracy under difference scenarios.

C. Experiment Setup

We compare the performance of GluGAN, i.e., the quality of the synthetic data, with a group of GAN models using temporal settings in the literature. Specifically, the considered baseline methods include TimeGAN [35], RCGAN [33], C-RNN-GAN [32], and WaveGAN [34]. Among these, TimeGAN, C-RNN-GAN, and WaveGAN use univariate glucose data, while GluGAN and RCGAN use multivariate inputs. For TimeGAN, only glucose time series is generated without the use of static features [35], so we apply the univariate setting. We retain the main architectures of the baseline GAN models and tune the hyperparameters by the same hold-out validation datasets, according to the predictive scores. To avoid model overfitting, we utilize early stopping to model training with a patience number of 50.

To evaluate the efficacy of data augmentation, three predictors, including the post-hoc LSTM, dilated RNN (DRNN) [25], and support vector regression (SVR) [48], are evaluated on the testing sets as described in Section II-D with 30 and 60-minute prediction horizons (PHs). DRNN is a state-of-the-art model to accurately predict glucose levels for the OhioT1DM dataset in our previous work [25], which is based on three dilated RNN layers. We also apply the same transfer-learning framework (Fig. 7b) to improve the performance of LSTM and DRNN predictors [25], [26]. SVR is a robust machine learning model in glucose prediction and is commonly used as a baseline method in the literature [22]. All the deep learning models are developed by TensorFlow 1.15 and Python 3.7, while SVR (RBF kernel) is deployed by scikit-learn 0.24. Training the deep neural networks is accelerated by NVIDIA GTX 1080 Ti GPU.

D. Results

Fig. 4 shows the results of PCA and t-SNE analysis for three T1D subjects from OhioT1DM, ARISES, and ABC4D testing sets, respectively. It is to be noted that the distributions of synthetic and real glucose time series are highly overlapped, indicating good similarity.

In order to quantitatively evaluate the quality of synthetic data, we compared the performance of GluGAN with four existing GAN frameworks. We computed the statistical significance (p-value) by paired t-test after confirming the normality of distributions by the Shapiro-Wilk test. Fig. 5a depicts the performance of discriminative scores for GluGAN and the considered baseline methods evaluated on the three clinical datasets. Notably, GluGAN achieved the smallest mean

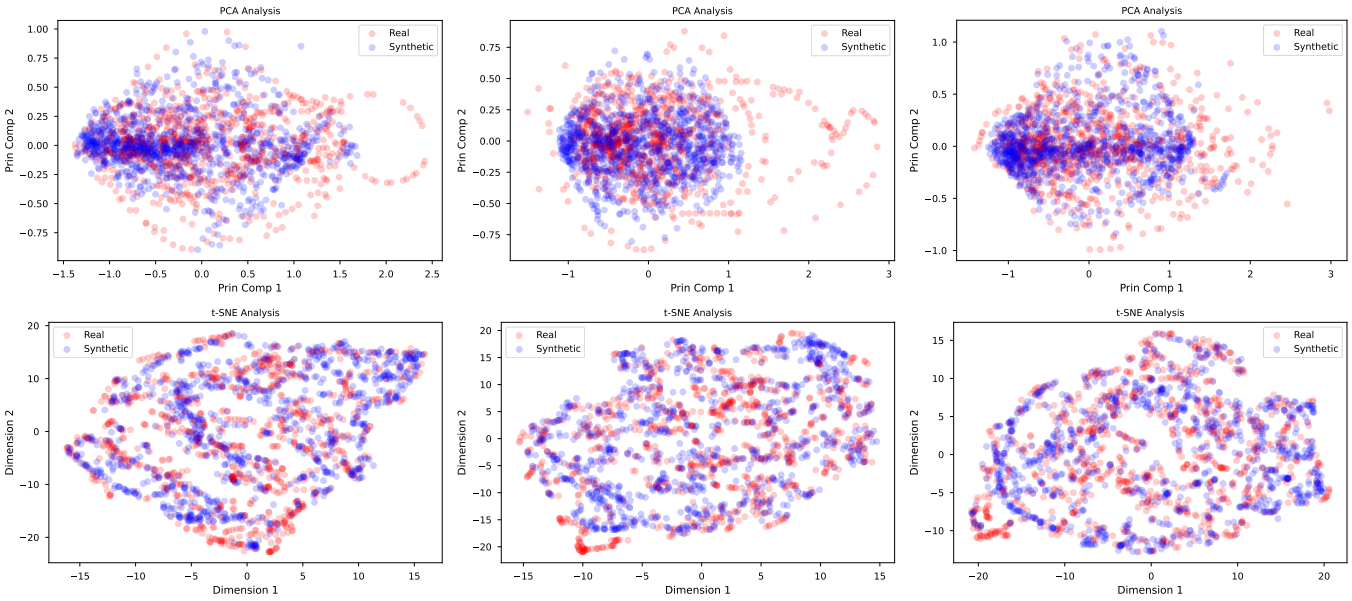

Fig. 4: PCA (the top row) and t-SNE (the bottom row) analysis on the distributions of real and synthetic glucose sequences that are displayed in the red and blue dots, respectively. The columns from left to right show the plots for the OhioT1DM, ARISES, and ABC4D datasets, respectively.

discriminative scores across all the datasets (0.17 ± 0.09, 0.16 ± 0.07, 0.13 ± 0.05; overall result: 0.15 ± 0.07) and obtained significant improvements, except for the OhioT1DM dataset, when compared with the baseline methods. Similarly, Fig. 5b shows the results of predictive scores. Compared with the considered baseline methods, GluGAN also significantly improved the performance and achieved better mean predictive scores of 6.29 ± 1.31, 6.08 ± 0.98, and 7.08 ± 1.26 mg/dL for the OhioT1DM, ARISES, and ABC4D datasets (overall result: 6.64 ± 1.30 mg/dL), respectively.

Fig. 5c and 5d show the MAE of TBR and TAR, where GluGAN accurately estimated the percentages of time spent in hypoglycemia and hyperglycemia. In terms of the distance metrics, EMD and MMD, the distribution of the synthetic data generated by GluGAN is closest to that of real data for all the datasets, as shown Fig. 5e and 5f.

Fig. 6 shows an example of a one-day period of real and synthetic glucose time series, i.e., ambulatory glucose profiles. It is observed that the synthetic curve passes through three of the four SMBG measurements and has trends and peaks that are highly correlated with the actual CGM measurements. In the experiments, we noted that excluding SMBG features did not have a significant impact on predictive scores but caused larger bias and degraded the overall mean discriminative scores by 0.02.

Table I presents the performance of glucose prediction with three data-driven prediction algorithms over 30 and 60-minute PHs. Each predictor is tested by TATR and TRTR routines. It is worth noting that the use of augmented training sets in TATR significantly reduced RMSE and MAE scores in each dataset.

IV. Discussion

To the best of our knowledge, this work is the first attempt to generate realistic T1D glucose time series based on a specifically designed GAN framework, i.e., GluGAN. To prove the validity and evaluate the performance of the approach, three clinical datasets were employed. The visualization in Fig. 4 and results in Fig. 5a demonstrate that GluGAN is able to generate high-fidelity synthetic glucose data, of which the distributions are similar to those of real data. The results in Fig. 5b indicate that the synthetic data preserved good temporal dynamics and can be useful in terms of glucose prediction tasks. It is to be noted that, by taking advantage of the autoregressive formulation and latent space auto-encoding, GluGAN and TimeGAN achieved much smaller discriminative and predictive scores than the other three baseline GAN models. We noticed that it is challenging to train the GAN model to generate glucose time series without supervised loss, mainly due to the high correlation cross timesteps (Fig. 3). The most significant difference between GluGAN and TimeGAN is the use of conditional inputs, which is inspired by RCGAN [33]. The conditional inputs provide additional information on glucose dynamics and contribute to improved performance, which is also the key module to enable *in silico* trials. Although TimeGAN is capable of generating static features, such as gender and age, it does not support dynamic conditional inputs as we do in GluGAN.

C-RNN-GAN used a feature-matching approach [42] with a supervised loss in generator training, aiming to match the hidden representations between real and synthetic data [32]. We also used this basic supervised loss in another baseline method, RCGAN; otherwise, the model would fail to generate realistic data. The authors of WaveGAN used the Wasserstein distance and a gradient penalty [49] to improve the loss of the

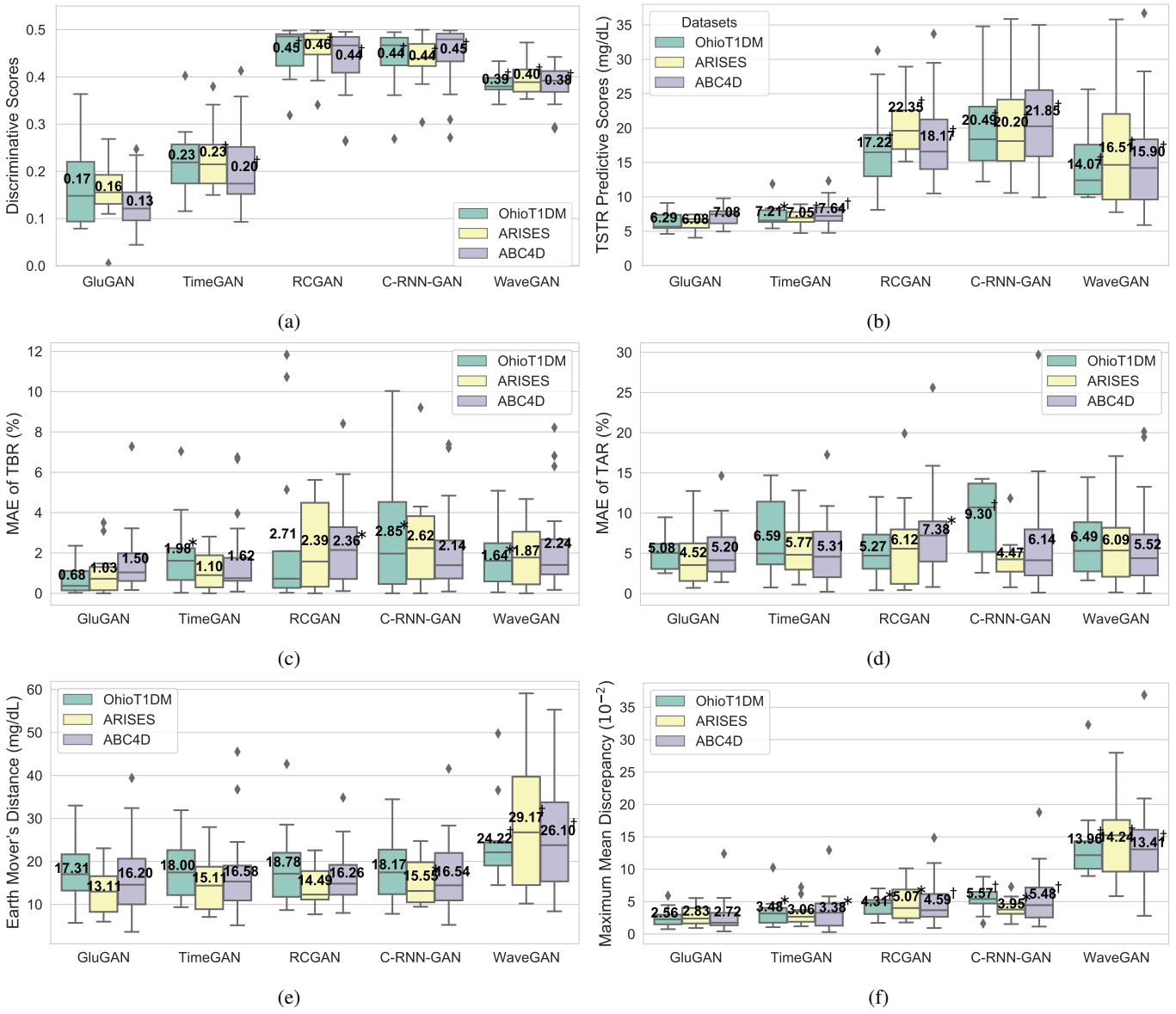

Fig. 5: Performance of GAN models evaluated on the OhioT1DM, ARISES, and ABC4D datasets. (a) Discriminative scores. (b) TSTR predictive scores. (c) MAE of TBR. (d) MAE of TAR. (e) Earth mover's distance. (f) Maximum mean discrepancy. The central lines of the boxplots indicate the median values, and the whiskers indicate the distance of 1.5 times the interquartile range. The mean values are shown in the center of the boxplots. Statistical significance is indicated as * for $p \leq 0.05$ and † for $p \leq 0.01$.

generator [34]. Furthermore, to investigate whether GluGAN simply memorized training data and reproduced them during the generative phase, we computed the MMD scores for synthetic training data ($\hat{\mathcal{D}}_{tr}$) and synthetic testing data ($\hat{\mathcal{D}}_{te}$) and employed the Kolmogorov-Smirnov test with the null hypothesis that these two groups of scores are sampled from the same distributions. In the experiments, we have $p > 0.05$ for all the datasets, indicating the performance on the training data is not significantly better than that on the testing data.

We also applied GluGAN to increase the amount of available training data, and thus to enhance the performance of a glucose prediction algorithm. Particularly, we explored two deep learning predictors and a machine learning predictor with 30 and 60-minute PHs. As observed in Table I, the use of augmented training sets reduced RMSE and MAE for all the predictors over the two PHs. In this case, GluGAN is an effective and model-agnostic solution to meet the challenge of limited personal data and the cold start issue for the development of data-driven models. Fig. 6 shows the visualization of synthetic glucose over 24 hours, which was obtained by retraining the GluGAN model with an input length of one day. It is worth noting that the trends and peaks of the synthetic data are similar to that of the real CGM measurements, which may offer an estimation of ambulatory glucose files for T1D subjects with the SMBG regimen only.

Meanwhile, we noted several limitations in this study. The

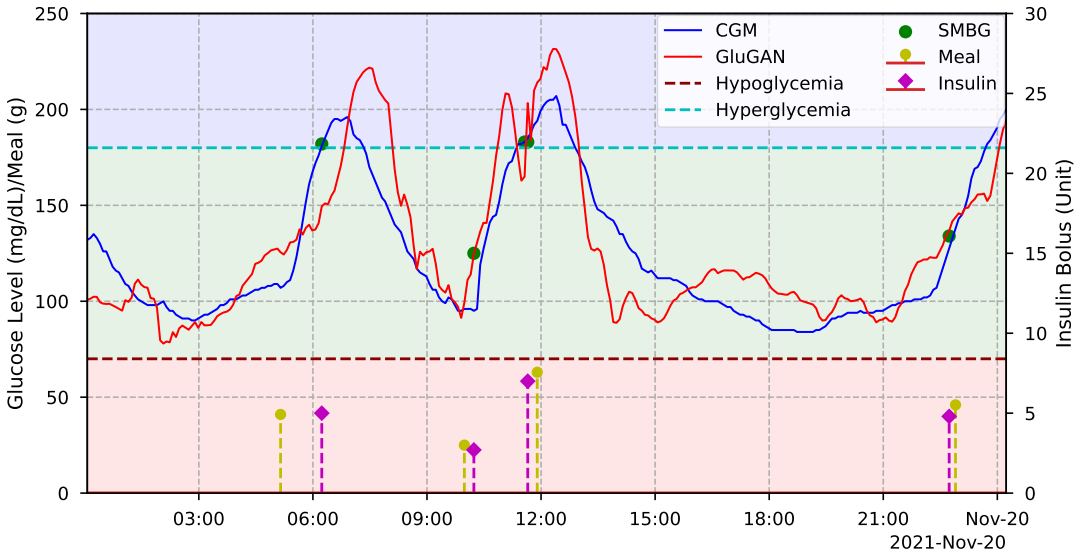

Fig. 6: Visualization of synthetic glucose time series for a T1D subject in the OhioT1DM dataset over a day. The real CGM readings and the synthetic glucose values generated by GluGAN are displayed in blue and red solid lines, respectively. The hypoglycemic, euglycemic, and hyperglycemic regions are marked by light red, green, and blue shaded areas, respectively. The conditional inputs of SMBG, carbohydrate of meal intake, and insulin bolus are shown in green dots, vertical yellow lines, and vertical magenta lines, respectively.

TABLE I: Performance of glucose prediction by LSTM, DRNN, and SVR trained on augmented (TATR) and original (TRTR) training sets. Statistical significance is indicated as * for $p \leq 0.05$ and † for $p \leq 0.01$.

Datasets	**Method**	**LSTM**		**DRNN**		**SVR**	
		PH = 30					
		RMSE	MAE	RMSE	MAE	RMSE	MAE
OhioT1DM	TATR	20.28 ± 2.42	14.42 ± 1.42	19.80 ± 2.35	14.10 ± 1.55	22.00 ± 5.26	15.01 ± 2.17
	TRTR	$21.30 \pm 2.43^\dagger$	$15.55 \pm 1.57^\dagger$	$20.39 \pm 2.48^\dagger$	$14.64 \pm 1.71^\dagger$	$23.89 \pm 4.51^\dagger$	$17.28 \pm 2.19^\dagger$
ARISES	TATR	21.57 ± 4.04	15.78 ± 2.88	21.02 ± 3.73	15.35 ± 2.63	24.61 ± 6.10	17.11 ± 3.76
	TRTR	$22.81 \pm 4.11^\dagger$	$16.86 \pm 2.97^\dagger$	21.69 ± 4.03	$16.01 \pm 2.98^*$	$26.39 \pm 7.38^\dagger$	$19.09 \pm 5.01^\dagger$
ABC4D	TATR	21.03 ± 2.47	14.42 ± 1.42	20.81 ± 2.45	15.07 ± 1.82	21.63 ± 2.15	15.59 ± 1.63
	TRTR	$22.08 \pm 2.45^\dagger$	$15.55 \pm 1.57^\dagger$	$21.21 \pm 2.52^\dagger$	$15.31 \pm 1.86^*$	$23.52 \pm 2.24^\dagger$	$17.87 \pm 1.95^\dagger$
Datasets	**Method**	PH= 60					
		RMSE	MAE	RMSE	MAE	RMSE	MAE
OhioT1DM	TATR	33.80 ± 3.64	25.05 ± 2.96	33.43 ± 3.56	24.80 ± 2.72	35.47 ± 5.99	25.59 ± 3.70
	TRTR	$35.61 \pm 4.50^*$	$26.64 \pm 3.61^*$	$34.38 \pm 3.73^*$	25.37 ± 2.97	$36.46 \pm 5.87^*$	$27.06 \pm 3.66^\dagger$
ARISES	TATR	37.10 ± 7.37	27.75 ± 5.41	36.54 ± 7.31	27.41 ± 5.44	39.10 ± 7.42	28.66 ± 5.62
	TRTR	$38.98 \pm 8.34^*$	$29.28 \pm 6.49^*$	38.19 ± 8.09	28.64 ± 6.22	39.23 ± 7.77	$29.31 \pm 6.17^*$
ABC4D	TATR	35.26 ± 4.89	26.37 ± 3.62	35.07 ± 4.85	26.28 ± 3.63	36.22 ± 4.76	26.74 ± 3.45
	TRTR	$36.37 \pm 4.95^\dagger$	$27.46 \pm 3.74^\dagger$	$36.03 \pm 4.90^\dagger$	$27.05 \pm 3.62^\dagger$	36.64 ± 4.50	$27.91 \pm 3.43^\dagger$

first is the lack of efficient and unified metrics to validate GAN models and tune hyperparameters. Previous studies used domain-specific metrics (e.g., tones of music) [32] and simple visual assessment [33], which are not suitable for our task. Considering the high ACF existing in glucose time series (Fig. 3), we used predictive scores (RMSE) in model validation, but it is time-consuming to retrain a post-hoc RNN whenever a model setting changes. In future work, we will investigate multiple statistic scores (such as cointegration tests) and clinical indicators (such as glycemic variability [50]) to develop comprehensive metrics to evaluate the quality of synthetic glucose data. Another limitation of GluGAN is that the model sometimes exhibits anomalous behaviors with certain conditional inputs, e.g., slow increase or even decrease of glucose levels after meal intake. Although these anomalous events are possible in actual clinical settings due to some external events, such as delayed meal absorption or physical exercise, it hinders GluGAN to become a reliable and trustworthy personalized *in silico* simulator to further assist decision support in T1D management. It is also possible that GluGAN learned to mimic the underlying real-life events. However, these events are rarely recorded by the T1D subjects and lack unified quantification methods, e.g., exercise reported by subjective assessment in the OhioT1DM datasets. In the future, we will introduce other time series inputs, such as vital signs measured by sensor wristbands, to better quantify these events. Basal insulin could also be a useful conditional input feature, especially for subjects with CSII treatment. We did not include this feature in the current work, since many T1D subjects in the ARISES and ABC4D datasets used MDI

therapy and thus did not provide information on basal insulin. Moreover, we plan to explore EMD loss function, i.e., Wasserstein distance, and transformer-based neural networks, aiming to further enhance model performance and consequently, the quality of the generated synthetic data.

To perform *in silico* trials to evaluate the efficacy of new treatments and interventions, the conditional inputs (i.e., insulin bolus doses and carbohydrates entries) are modified, and the corresponding glucose levels are simulated with GluGAN. Future work includes simulating the conditional inputs using the two following methods. One consists in simulating these events from a predefined daily pattern with additional randomness on meal size and meal timing, like the functionality included in the UVA/Padova T1D simulator. The other method consists in using kernel density estimation to create a personalized daily pattern for each individual and then generate new samples by sampling from the resulting distribution. To develop a fully-functional data-driven T1D simulators, future work also includes increasing the interpretability and causality of deep learning models, for which the potential methods include casual graphs [51] and exponential objective functions [52]. Furthermore, we consider combining physiological models and CGM models (e.g, the models in the UVA/Padova T1D simulator [28]) with GluGAN to obtain robust generative performance and finally develop personalized T1D digital twins. In this case, it will initiate revolutionary changes in both pre-clinical and clinical T1D studies, which allows clinicians to freely test and adjust treatment and daily management policy in a virtual environment. Training with properly masked inputs [53], we will also explore the feasibility of applying GluGAN to impute missing gaps of CGM data.

V. Conclusion

In this work, we proposed a novel framework (GluGAN) based on GAN architectures and deep learning technologies for the generation of realistic blood glucose time series in T1D. In the reported experiments, we demonstrated that GluGAN is able to generate high-quality synthetic data and outperformed all the considered baseline GAN models. As an application of the proposed approach, we used GluGAN to increase the size of training data as a data augmentation technique for blood glucose prediction. In particular, we enhanced the prediction accuracy of different machine learning-based glucose predictors (LSTM, DRNN, and SVR) over 30 and 60-minute PHs. The promising results of this work have demonstrated the feasibility of using GluGAN to improve data-driven decision support systems in T1D management. Finally, in combination with physiological models and clinical constraints, GluGAN has the potential to be employed as a personalized T1D simulator or a digital twin in future work.

Appendix

A. Hyperparameters

Table II lists the hyperparameters used in this work.

B. Cohort Characteristics

Table III summarizes the demographic characteristics of the T1D subjects in the three clinical datasets.

TABLE II: List of hyperparameters.

Parameter	Value
Length of glucose time series L	24
Iterations of embedding learning $\mathcal{T}_R$	10,000
Iterations of supervised learning $\mathcal{T}_S$	10,000
Iterations of joint learning $\mathcal{T}_J$	25,000
Ratio of reconstruction loss λ_1	1
Ratio of unsupervised loss λ_2	10
Number of the inner loop steps k	2
Batch size	128
Threshold of discriminator loss l_D	0.15
Hidden units of the RNN layer	64
Learning rate of the Adam optimizer	0.001
Early stopping patience	50

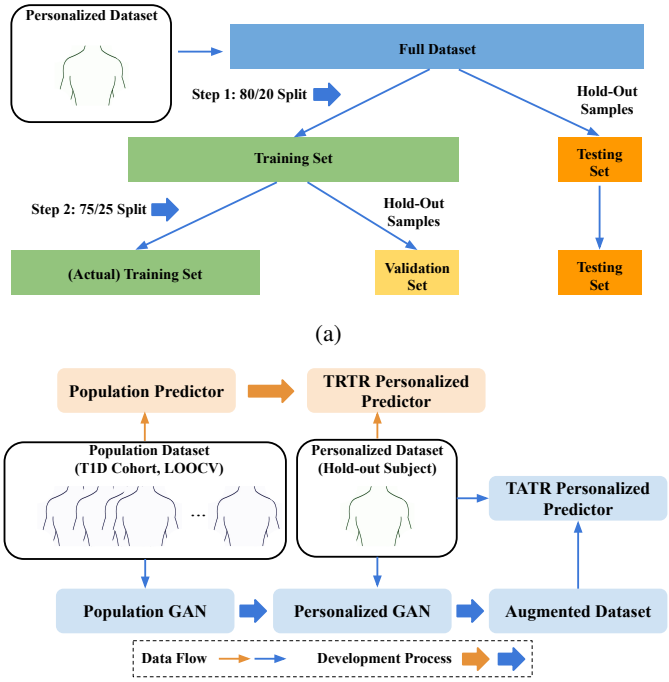

Fig. 7: Data split and experiment setup. (a) Data split for developing all the considered GAN models and evaluating the quality of synthetic data. (b) Diagram of the TRTR (orange arrows) and TATR (blue arrows) in data augmentation experiments for glucose prediction. The transfer-learning framework with LOOCV is applied to develop personalized GluGAN and deep learning-based predictors (i.e., LSTM and DRNN), where only two-week data are available. The thin and thick arrows indicate data flow and development process, respectively.

C. Data Generation and Augmentation Experiments

Fig. 7 illustrates the experiment setup of evaluating the quality of synthetic data and performing data augmentation for glucose prediction.

Acknowledgment

This research has been funded by Engineering and Physical Sciences Research Council (EPSRC EP/P00993X/1) and the President's PhD Scholarship at Imperial College London (UK). We would like to thank Prof. Nick Oliver, Dr. Monika

TABLE III: Demographic characteristics (Median(IQR)) of the OhioT1DM, ARISES, and ABC4D datasets

	OhioT1DM	ARISES	ABC4D
Age	50.0 (40.0-60.0)*	40.0 (30.0-49.0)	36.0 (29.0–46.0)
Gender (female/male)	5/7 (41.7% female)	6/6 (50.0% female)	15/25 (60.0% female)
Insulin treatment (CSII/MDI)	12/0 (100.0 % CSII)	6/6 (50.0% CSII)	8/17 (47.1% CSII)
HbA1c (mmol/mol)	N/A	50.4 (41.5-57.5)	61.0 (52.0–66.0)
Mean glucose level (mg/dL)	153.1 (20.1)	163.7 (43.2)	159.9 (38.6)
Time below range (%)	2.6 (3.0)	2.9 (2.6)	3.5 (5.2)
Time in range (%)	64.8 (10.5)	62.5 (23.4)	62.1 (23.8)
Time above range (%)	29.9 (13.8)	36.8 (26.0)	32.8 (21.8)
Low blood glucose index	0.8 (0.7)	0.7 (0.6)	1.1 (1.2)
Inter-day coefficient of variation (%)	36.5 (6.3)	35.2 (4.1)	36.2 (8.0)
Intra-day coefficient of variation (%)	30.8 (5.8)	30.3 (4.1)	30.7 (8.2)

*De-identified data.

Reddy, Dr. Chukwuma Uduku, and Narvada Jugnee for their contribution in obtaining the ABC4D and ARISES datasets.

REFERENCES

[1] P. Saeedi, I. Petersohn, P. Salpea, B. Malanda, S. Karuranga, N. Unwin, S. Colagiuri, L. Guariguata, A. A. Motala, K. Ogurtsova *et al.*, "Global and regional diabetes prevalence estimates for 2019 and projections for 2030 and 2045: Results from the international diabetes federation diabetes atlas," *Diabetes Research and Clinical Practice*, vol. 157, p. 107843, 2019.

[2] E. W. Gregg, N. Sattar, and M. K. Ali, "The changing face of diabetes complications," *The Lancet Diabetes & Endocrinology*, vol. 4, no. 6, pp. 537–547, 2016.

[3] E. M. Benjamin, "Self-monitoring of blood glucose: the basics," *Clinical Diabetes*, vol. 20, no. 1, pp. 45–47, 2002.

[4] Juvenile Diabetes Research Foundation Continuous Glucose Monitoring Study Group, "Continuous glucose monitoring and intensive treatment of type 1 diabetes," *New England Journal of Medicine*, vol. 359, no. 14, pp. 1464–1476, 2008.

[5] J. Davidson, "Strategies for improving glycemic control: effective use of glucose monitoring," *The American Journal of Medicine*, vol. 118, no. 9, pp. 27–32, 2005.

[6] S. Clarke and J. Foster, "A history of blood glucose meters and their role in self-monitoring of diabetes mellitus," *British Journal of Biomedical Science*, vol. 69, no. 2, pp. 83–93, 2012.

[7] S. R. Patton and M. A. Clements, "Continuous glucose monitoring versus self-monitoring of blood glucose in children with type 1 diabetes-are there pros and cons for both?" *US Endocrinology*, vol. 8, no. 1, p. 27, 2012.

[8] B. Floyd, P. Chandra, S. Hall, C. Phillips, E. Alema-Mensah, G. Strayhorn, E. O. Ofili, and G. E. Umpierrez, "Comparative analysis of the efficacy of continuous glucose monitoring and self-monitoring of blood glucose in type 1 diabetes mellitus," *Journal of Diabetes Science and Technology*, vol. 6, no. 5, pp. 1094–1102, 2012.

[9] A. Facchinetti, "Continuous glucose monitoring sensors: past, present and future algorithmic challenges," *Sensors*, vol. 16, no. 12, p. 2093, 2016.

[10] G. Cappon, G. Acciaroli, M. Vettoretti, A. Facchinetti, and G. Sparacino, "Wearable continuous glucose monitoring sensors: a revolution in diabetes treatment," *Electronics*, vol. 6, no. 3, p. 65, 2017.

[11] G. Cappon, M. Vettoretti, G. Sparacino, and A. Facchinetti, "Continuous glucose monitoring sensors for diabetes management: a review of technologies and applications," *Diabetes & Metabolism Journal*, vol. 43, no. 4, pp. 383–397, 2019.

[12] P. Rossetti, J. Bondia, J. Vehí, and C. G. Fanelli, "Estimating plasma glucose from interstitial glucose: the issue of calibration algorithms in commercial continuous glucose monitoring devices," *Sensors*, vol. 10, no. 12, pp. 10 936–10 952, 2010.

[13] S. Kitsiou, G. Paré, M. Jaana, and B. Gerber, "Effectiveness of mhealth interventions for patients with diabetes: an overview of systematic reviews," *PloS one*, vol. 12, no. 3, p. e0173160, 2017.

[14] A. Menychtas, P. Tsanakas, and I. Maglogiannis, "Knowledge discovery on IoT-enabled mhealth applications," in *GeNeDis 2018*. Springer, 2020, pp. 181–191.

[15] T. Zhu, L. Kuang, K. Li, J. Zeng, P. Herrero, and P. Georgiou, "Blood glucose prediction in type 1 diabetes using deep learning on the edge," in *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2021, pp. 1–5.

[16] M. Vettoretti, G. Cappon, A. Facchinetti, and G. Sparacino, "Advanced diabetes management using artificial intelligence and continuous glucose monitoring sensors," *Sensors*, vol. 20, no. 14, p. 3870, 2020.

[17] I. Kavakiotis, O. Tsave, A. Salifoglou, N. Maglaveras, I. Vlahavas, and I. Chouvarda, "Machine learning and data mining methods in diabetes research," *Computational and Structural Biotechnology Journal*, vol. 15, pp. 104–116, 2017.

[18] T. Zhu, K. Li, P. Herrero, and P. Georgiou, "Basal glucose control in type 1 diabetes using deep reinforcement learning: An in silico validation," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 4, pp. 1223–1232, 2021.

[19] T. Zhu, K. Li, L. Kuang, P. Herrero, and P. Georgiou, "An insulin bolus advisor for type 1 diabetes using deep reinforcement learning," *Sensors*, vol. 20, no. 18, p. 5058, 2020.

[20] J. Beauchamp, R. Bunescu, C. Marling, Z. Li, and C. Liu, "Lstms and deep residual networks for carbohydrate and bolus recommendations in type 1 diabetes management," *Sensors*, vol. 21, no. 9, p. 3303, 2021.

[21] M. Muñoz-Organero, P. Queipo-Álvarez, and B. García Gutiérrez, "Learning carbohydrate digestion and insulin absorption curves using blood glucose level prediction and deep learning models," *Sensors*, vol. 21, no. 14, p. 4926, 2021.

[22] T. Zhu, K. Li, P. Herrero, and P. Georgiou, "Deep learning for diabetes: A systematic review," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 7, pp. 2744–2757, 2021.

[23] T. Zhu, K. Li, P. Herrero, J. Chen, and P. Georgiou, "A deep learning algorithm for personalized blood glucose prediction," in *The 3rd International Workshop on Knowledge Discovery in Healthcare Data in the 27th IJCAI-ECAI*, 2018, pp. 74–78.

[24] K. Li, C. Liu, T. Zhu, P. Herrero, and P. Georgiou, "GluNet: A deep learning framework for accurate glucose forecasting," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 2, pp. 414–423, 2020.

[25] T. Zhu, K. Li, J. Chen, P. Herrero, and P. Georgiou, "Dilated recurrent neural networks for glucose forecasting in type 1 diabetes," *Journal of Healthcare Informatics Research*, vol. 4, no. 3, pp. 308–324, 2020.

[26] H. Rubin-Falcone, I. Fox, and J. Wiens, "Deep residual time-series forecasting: Application to blood glucose prediction," in *The 5th International Workshop on Knowledge Discovery in Healthcare Data, ECAI 2020*, 2020, pp. 105–109.

[27] G. Cappon, L. Meneghetti, F. Prendin, J. Pavan, G. Sparacino, S. Del Favero, and A. Facchinetti, "A personalized and interpretable deep learning based approach to predict blood glucose concentration in type 1 diabetes." in *The 5th International Workshop on Knowledge Discovery in Healthcare Data, ECAI 2020*, 2020, pp. 75–79.

[28] C. Dalla Man, F. Micheletto, D. Lv, M. Breton, B. Kovatchev, and C. Cobelli, "The uva/padova type 1 diabetes simulator: new features," *Journal of Diabetes Science and Technology*, vol. 8, no. 1, pp. 26–34, 2014.

[29] R. Visentin, C. Dalla Man, and C. Cobelli, "One-day bayesian cloning of type 1 diabetes subjects: toward a single-day uva/padova type 1 diabetes simulator," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 11, pp. 2416–2424, 2016.

[30] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.

[31] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019.

[32] O. Mogren, "C-RNN-GAN: A continuous recurrent neural network with adversarial training," in *Constructive Machine Learning Workshop (CML) at NIPS 2016*, 2016, p. 1.
[33] C. Esteban, S. L. Hyland, and G. Rätsch, "Real-valued (medical) time series generation with recurrent conditional GANs," *CoRR*, vol. abs/1706.02633, 2017. [Online]. Available: http://arxiv.org/abs/1706.02633
[34] C. Donahue, J. McAuley, and M. Puckette, "Adversarial audio synthesis," in *International Conference on Learning Representations*, 2018.
[35] J. Yoon, D. Jarrett, and M. van der Schaar, "Time-series generative adversarial networks," *Advances in Neural Information Processing Systems*, vol. 32, pp. 5508–5518, 2019.
[36] Y. Deng, L. Lu, L. Aponte, A. M. Angelidi, V. Novak, G. E. Karniadakis, and C. S. Mantzoros, "Deep transfer learning and data augmentation improve glucose levels prediction in type 2 diabetes patients," *npj Digital Medicine*, vol. 4, no. 1, pp. 1–13, 2021.
[37] T. Zhu, X. Yao, K. Li, P. Herrero, and P. Georgiou, "Blood glucose prediction for type 1 diabetes using generative adversarial networks," in *The 5th International Workshop on Knowledge Discovery in Healthcare Data, ECAI 2020*, 2020, pp. 90–94.
[38] S. Boschert and R. Rosen, "Digital twin—the simulation aspect," in *Mechatronic futures*. Springer, 2016, pp. 59–74.
[39] H. Elayan, M. Aloqaily, and M. Guizani, "Digital twin for intelligent context-aware IoT healthcare systems," *IEEE Internet of Things Journal*, 2021.
[40] T. Siegmund, L. Heinemann, R. Kolassa, and A. Thomas, "Discrepancies between blood glucose and interstitial glucose—technological artifacts or physiology: implications for selection of the appropriate therapeutic target," *Journal of Diabetes Science and Technology*, vol. 11, no. 4, pp. 766–772, 2017.
[41] F. Huszár, "How (not) to train your generative model: Scheduled sampling, likelihood, adversary?" *arXiv preprint arXiv:1511.05101*, 2015.
[42] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," *Advances in neural information processing systems*, vol. 29, pp. 2234–2242, 2016.
[43] C. Marling and R. Bunescu, "The OhioT1DM dataset for blood glucose level prediction: Update 2020," in *The 5th KDH workshop, ECAI 2020*, 2020, pp. 71–74.
[44] P. Herrero, A. Alalitei, M. Reddy, P. Georgiou, and N. Oliver, "Robust determination of the optimal continuous glucose monitoring length of intervention to evaluate long-term glycemic control," *Diabetes Technology & Therapeutics*, vol. 23, no. 4, pp. 314–319, 2021.
[45] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4, pp. 433–459, 2010.
[46] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE." *Journal of Machine Learning Research*, vol. 9, no. 11, 2008.
[47] T. Battelino, T. Danne, R. M. Bergenstal, S. A. Amiel, R. Beck, T. Biester, E. Bosi, B. A. Buckingham, W. T. Cefalu, K. L. Close *et al.*, "Clinical targets for continuous glucose monitoring data interpretation: recommendations from the international consensus on time in range," *Diabetes Care*, vol. 42, no. 8, pp. 1593–1603, 2019.
[48] E. I. Georga, V. C. Protopappas, D. Ardigo, M. Marina, I. Zavaroni, D. Polyzos, and D. I. Fotiadis, "Multivariate prediction of subcutaneous glucose concentration in type 1 diabetes patients based on support vector regression," *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 1, pp. 71–81, 2012.
[49] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of Wasserstein GANs," in *Advances in Neural Information Processing Systems*, 2017.
[50] A. H. El-Laboudi, I. F. Godsland, D. G. Johnston, and N. S. Oliver, "Measures of glycemic variability in type 1 diabetes and the effect of real-time continuous glucose monitoring," *Diabetes Technology & Therapeutics*, vol. 18, no. 12, pp. 806–812, 2016.
[51] M. Kocaoglu, C. Snyder, A. G. Dimakis, and S. Vishwanath, "Causalgan: Learning causal implicit generative models with adversarial training," in *International Conference on Learning Representations*, 2018.
[52] R. She, P. Fan, X.-Y. Liu, and X. Wang, "Interpretable generative adversarial networks with exponential function," *IEEE Transactions on Signal Processing*, 2021.
[53] Y. Luo, X. Cai, Y. Zhang, J. Xu, and X. Yuan, "Multivariate time series imputation with generative adversarial networks," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 1603–1614.