




GluNet: A Deep Learning Framework for Accurate Glucose Forecasting

Kezhi Li , Chengyuan Liu , Taiyu Zhu, Pau Herrero, and Pantelis Georgiou 

Abstract—For people with Type 1 diabetes (T1D), forecasting of blood glucose (BG) can be used to effectively avoid hyperglycemia, hypoglycemia and associated complications. The latest continuous glucose monitoring (CGM) technology allows people to observe glucose in real-time. However, an accurate glucose forecast remains a challenge. In this work, we introduce GluNet, a framework that leverages on a personalized deep neural network to predict the probabilistic distribution of short-term (30–60 minutes) future CGM measurements for subjects with T1D based on their historical data including glucose measurements, meal information, insulin doses, and other factors. It adopts the latest deep learning techniques consisting of four components: data pre-processing, label transform/recover, multi-layers of dilated convolution neural network (CNN), and post-processing. The method is evaluated *in-silico* for both adult and adolescent subjects. The results show significant improvements over existing methods in the literature through a comprehensive comparison in terms of root mean square error (RMSE) (8.88 ± 0.77 mg/dL) with short time lag (0.83 ± 0.40 minutes) for prediction horizons (PH) = 30 mins (minutes), and RMSE (19.90 ± 3.17 mg/dL) with time lag (16.43 ± 4.07 mins) for PH = 60 mins for virtual adult subjects. In addition, GluNet is also tested on two clinical data sets. Results show that it achieves an RMSE (19.28 ± 2.76 mg/dL) with time lag (8.03 ± 4.07 mins) for PH = 30 mins and an RMSE (31.83 ± 3.49 mg/dL) with time lag (17.78 ± 8.00 mins) for PH = 60 mins. These are the best reported results for glucose forecasting when compared with other methods including the neural network for predicting glucose (NNPG), the support vector regression (SVR), the latent variable with exogenous input (LVX), and the auto regression with exogenous input (ARX) algorithm.

Index Terms—Deep learning, dilated convolutions, glucose forecasting, continuous glucose monitoring (CGM), diabetes.

I. INTRODUCTION

DIABETES is a chronic disease that affects about 415 million people around the world [1]. There are two main types of diabetes: Type-1 diabetes (T1D) and Type-2 diabetes (T2D). For people with T1D, exogenous insulin

administration is required to manage BG levels in the target range (70–180 mg/dl) to avoid hypoglycemia (<70 mg/dl) or hyperglycemia (>180 mg/dl). The recent developments in CGM technology [2] provide real-time monitoring of the current glucose, so people can observe their glucose concentration trend visually on devices (Fig. 1). It has been proven that CGM technology helps T1D subjects to have a better glycemic control [3], [4] by alerting them when hypo- or hyperglycemia events occur [5]–[7].

In recent years many algorithms have been proposed for glucose forecasting [8], such as polynomial models (autoregressive (AR), autoregressive exogenous (ARX), autoregressive moving-average (ARMA)) [9], machine learning models [10] and latent-variable-based statistical models [11]. Oviedo *et al.* [8] reviewed the topic comprehensively. In addition, machine learning (ML) techniques were extensively exploited, including support vector regression (SVR) [12], random forest (RF) [13] and grammatical evolution [14]. Deep learning (DL) is a promising ML technique which is still relatively new in this field. DL has potential because it can capture the complex dynamics of systems, particularly when it is difficult to derive the mathematical expressions of the system. DL has improved the state-of-the-art performance significantly in computer vision (CV) and bio-engineering applications [15]. There are a few works to date using DL for glucose forecasting. For instance, in [10], [16] traditional dense neural networks were used but without exploiting the advantages of deep layers. In [17], two latent layer neural networks were adopted for hypo/hyperglycemia prediction. However, this work lacks a comprehensive comparison with other algorithms.

In this paper we introduce GluNet, a framework that leverages deep neural networks (DNN) for forecasting accurate short-time CGM measurements (PH = 30 minutes and 60 minutes) using life-style data. These data include historical BG data from CGM, meal intake, insulin dosage. Other data from wearable technology or physiological glucose-insulin regulatory models are optional. GluNet consists of several components, including data preprocessing, label transform/recover, multi-layers of dilated convolution neural network (CNN), gated activations, residual and skip connections. The original idea comes from *PixelCNN* [18] and *Wavenet* [19], which are DL models used in CV and acoustic signal processing. Dilated CNNs are good at processing multi-dimensional long signals with a wider receptive field, and gated activation units capture the non-linearity of the probabilistic relations among several inputs. We modify *PixelCNN/Wavenet* accordingly to make it appropriate for glucose forecasting, and the DNN becomes a predictive model with fast

Manuscript received February 15, 2019; revised June 13, 2019, July 10, 2019, and July 24, 2019; accepted July 25, 2019. Date of publication July 29, 2019; date of current version February 6, 2020. This work was supported by EPSRC EP/P00993X/1. (Corresponding author: Kezhi Li.)

The authors are with the Department of Electronic and Electrical Engineering, Imperial College London, London SW7 2AZ, U.K. (e-mail: kezhi.li@imperial.ac.uk; chengyuan.liu@nottingham.ac.uk; taiyu.zhu17@imperial.ac.uk; p.herrero-vinias@imperial.ac.uk; pantelis@imperial.ac.uk).

Digital Object Identifier 10.1109/JBHI.2019.2931842

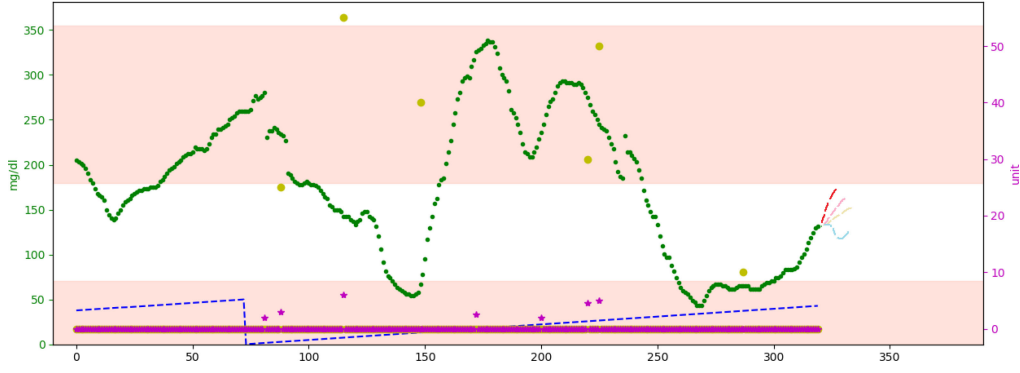


Fig. 1. An illustration of clinical CGM glucose time series and CGM measurements forecasting. x -axis stands for glycemic samples; y -left-axis represents the blood glucose level (in mg/dl). Green dots are glucose measurements sampled every 5 minutes by CGM in real-time. Purple dots are the insulin dosage (in units) shown on y -right-axis. Yellow dots denote the food carbohydrate (in gram) shown on the y -right-axis. The blue zigzag line represents the scale of one day length starting from the lowest point (0:00) to the highest point (24:00). The pink area means the hypoglycemia field (<70 mg/dl) and hyperglycemia field (>180 mg/dl). Dash lines at the right end of the green dots demonstrate several future forecasting glucose curves.

implementation instead of a generative model. GluNet offers a personalized glucose forecast, and the model can evolve when new personal data are collected. We have evaluated GluNet on 10 simulated virtual T1D adult and adolescent subjects, respectively, generated from the UVA/Padova T1D simulator [20], as well as two clinical dataset including 10 T1D subjects from the ABC4D project using Dexcom CGM sensors [21], and 6 T1D subjects from Ohio Dataset [22]. The results show that the proposed framework achieves state-of-the-art glucose forecasting in terms of accuracy and time lag on both *in silico* and the clinical dataset.

II. GLUNET

A. Problem Formulation

If a d -dimensional $\mathbf{x}_T = [\mathbf{x}_1, \dots, \mathbf{x}_T]^T \in \mathbb{R}^{T \times d}$ is the concatenation of historical data and features, where $\mathbf{x}_t = [\mathbf{m}_t, \mathbf{n}_t]$ consists of d_1 -dimensional historical data and d_2 -dimensional features,

$$\begin{aligned} \mathbf{m}_t &= [m_t(1), m_t(2), \dots, m_t(d_1)] \in \mathbb{R}^{d_1}, \\ \mathbf{n}_t &= [n_t(1), n_t(2), \dots, n_t(d_2)] \in \mathbb{R}^{d_2}, \end{aligned} \quad (1)$$

where d_2 -dimensional features \mathbf{n}_t are computed from \mathbf{m}_t , $d = d_1 + d_2$, and $m_t(1) \triangleq G_t$ is the CGM data at time t . Then the joint probability of the forecasting glucose, $p(\mathbf{G}')$, is factorised as a product of conditional probabilities as follows

$$p(\mathbf{G}') = \prod_{t=1}^{T-w} p(G'_{t+w} | G_1, \dots, G_t), \quad (2)$$

where w represents the number of glycemic samples in terms of the PH, and the forecasting glucose sample G'_{t+w} is therefore conditioned on the glucose samples at all previous timesteps. Specifically, the CGM measures the glucose every 5 minutes. If we forecast the glucose of next 30 minutes indicating PH = 30, then we set $w = 30/5 = 6$ samples. Note that 30 mins PH has been widely used in the literature [8] and applied in the Medtronic MiniMed 640G insulin pump. Because Pi (plasma

insulin estimation) and Ra (glucose rate of appearance) relate to glucose dynamics \mathbf{G} in addition to meal \mathbf{M} and insulin \mathbf{I} , for instance, we may set $d_1 = 3, d_2 = 2$ where $\mathbf{m}_t \triangleq [G_t, M_t, I_t]$ and $\mathbf{n}_t \triangleq [Pi_t, Ra_t]$, where $\{Pi_t, Ra_t\}$ can be calculated from $\{G_t, M_t, I_t\}$ and other information of subjects such as body weight [23]. The factors in consideration in \mathbf{x} can be changed accordingly, depending on the data we can collect.

In both training and inference, multi-dimensional health data are sent to the network as batches determined by a sliding window. GluNet uses a stack of dilated convolutional layers to model the conditional probability distribution. We set the outputs of GluNet as a classification over the next forecasting value G'_{t+w} targeting at the value G_{t+w} with a softmax layer. It is to maximize the log-likelihood of the data with regard to the parameters

$$\max \log \left\{ \prod_{t=1}^{T-w} p(G_{t+w} | \mathbf{x}_1, \dots, \mathbf{x}_t) \right\}. \quad (3)$$

After the training phase, the forecasting glucose G'_{t+w} 's distribution is computed for inference. We tune the hyperparameters on a validation set to avoid overfitting or underfitting carefully.

B. The Architecture of GluNet

GluNet has four components. They are preprocessing, label transforms and recover, multi-layers of dilated convolutions, postprocessing. In the preprocessing, we clean the data \mathbf{m}_t , compute useful \mathbf{n}_t , and prepare them for the DNN use. The label transform component converts the glucose G to the difference ΔG between current glucose G_t and PH glucose G_{t+w} . While in the inference, forecasting glucose is recovered from ΔG . The main ingredients of GluNet are multi-layers of dilated convolutions and gated activations. A stack of dilated convolutional layers are placed. Both residual and parameterised skip connections are applied throughout the network aiming at speeding up the convergence. In addition, we calculate the dilated CNN in a faster manner by eliminating redundant operations. Fig. 2 shows the architecture of GluNet.

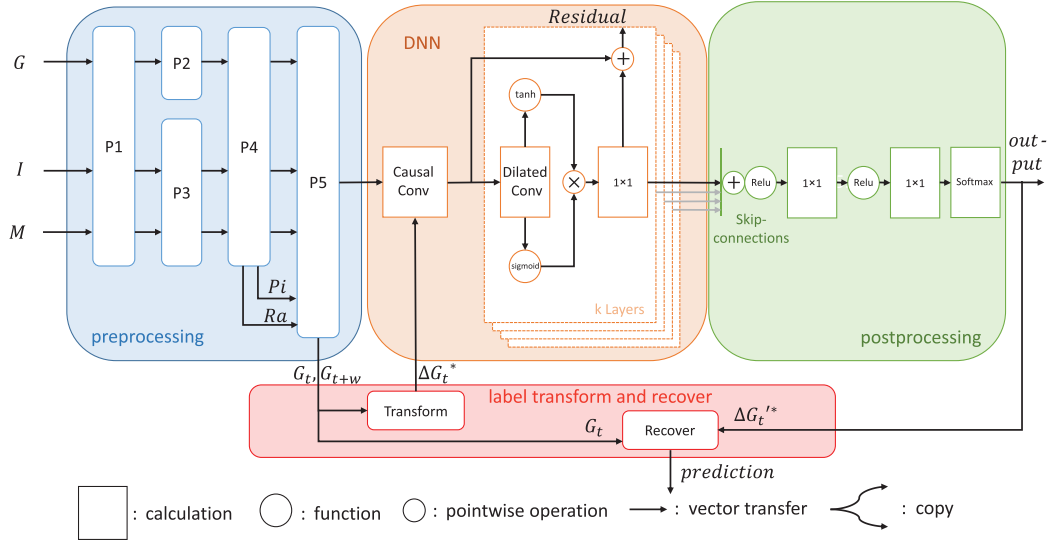


Fig. 2. The architecture of GluNet. It consists of four parts: preprocessing, DNN, postprocessing and label transform and recover. The input data are CGM measurements time series G , insulin I and meal M . More input factors are optional. In the preprocessing, P1: rule out outliers in G, I, M ; P2: interpolation of G when the missing data gap is not large; P3: fill or estimate the missing data in I, M ; P4: calculate other factors as input to the DNN, for instance plasma insulin estimation P_i and glucose rate of appearance R_a ; P5: align all factors with the same timeline and use them as input to the DNN. The aligned BG time series G_t is also sent to the label transform, and quantized ΔG_t^* is used as the category target in training.

C. Preprocessing

A preprocessing component is designed to clean the input data. There are several steps, P1: ruling out outliers, P2, P3: interpolation/extrapolation, P4: computing features, P5: alignment. The specific approaches of preprocessing depend on the specific dataset we have. 1) *Rule out outliers*: The outliers come from the error in CGM measurements, the transmission of CMG data, or incorrect behaviours when people record the data (meal, insulin). These outlier data points are supposed to be ruled out. In GluNet we set thresholds in the highest, lowest, and largest differential of the CGM measurements. Sometimes certain meal or insulin values in M or I are too large or too small, which implies that they might be mistakes. We identify these values by threshold method and correct them appropriately. 2) *Interpolation/extrapolation*: Spline interpolation or extrapolation technique are used when the missing CGM data are less than one hour (12 samples). Spline interpolation is implemented because it is more accurate, and simpler than some other methods [24]. If the missing CMG data are longer than 1 hour, we consider it as a separate dataset in training. In the inference, extrapolation is adopted because future samples cannot be used in forecasting. 3) *Considering other factors (optional)*: Given $\{I, M\}$, we utilize the insulin absorption model and the glucose absorption model proposed in [23] to compute R_a and P_i based on the estimation technique proposed by Hovorka in [25], where R_a denotes the glucose rate of appearance, P_i represents the plasma insulin estimation. More factors can be incorporated optionally. 4) *Alignment*: All data are aligned to G to generate multi-channel data that have the same length of G .

D. Label Transform and Recover

Aiming at increasing the predictive accuracy, a label transform component is placed before DNN. Rather than directly

forecasting G_{t+w} we forecast the difference between G_{t+w} and G_t denoted as $\Delta G_t = G_{t+w} - G_t$ in the training. An index of quantized ΔG_t , denoted as ΔG_t^* , is used as the output label. Without loss of generality the dimension of the output labels is set as 256. It represents that ΔG_t varies from -12.7 to 12.8 mg/dl, where each label denotes a quantized index ΔG_t^* with a quantizing interval of 0.1 mg/dl. Specifically, the label index ΔG_t^* can be calculated as

$$\Delta G_t^* = \begin{cases} 1 & \text{for } \Delta G_t < -12.75 \\ \mathcal{R}(10(\Delta G_t + 12.8)) & \text{for } -12.75 < \Delta G_t < 12.85 \\ 256 & \text{for } \Delta G_t > 12.85 \end{cases} \quad (4)$$

where \mathcal{R} denotes a round quantization operator that converts a decimal to an integer. In this way the predictive task is converted to a classification task: given previous data, the model computes the forecasting probability distribution of ΔG_t . GluNet forecasts the future glucose as the inference of the DNN by recovering BG values from label indexes. It is a reverse process of the previous label transform. Firstly $\Delta G_t'$ is computed using a reverse equation of (4) from inference value $\Delta G_t'^*$. Then the forecast G_{t+w}' can be derived from $G_{t+w}' = \Delta G_t' + G_t$ where G_t is the current glucose.

E. Dilated CNN

CNN is a feed-forward artificial neural network, and the calculation that performs in each layer is convolution. In GluNet we use the multi-layer convolutional NN to build the model. A visualization of a stack of causal convolutional layers is shown in Fig. 3 (I), where the forecast at timestep t depends on previous inputs. In PixelCNN, the convolution operator is equivalent to implementing a mask on image pixels by calculating an element-wise multiplication of the mask with the convolution kernel. For time series one can implement it by shifting output of normal

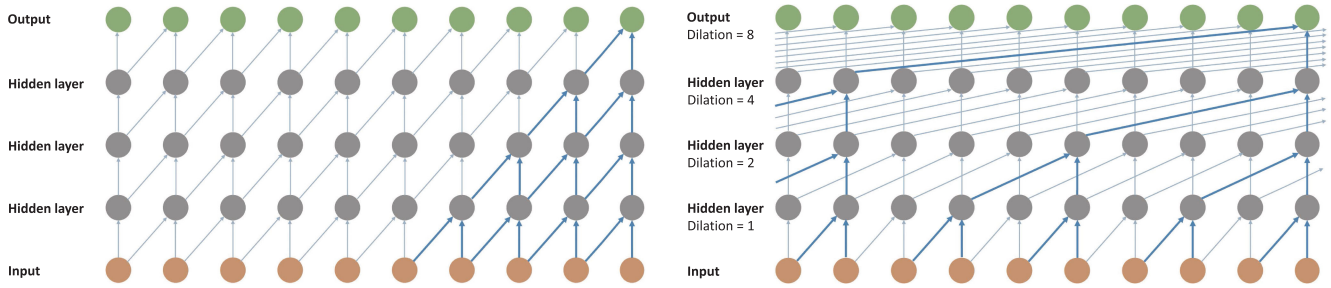


Fig. 3. Visualization of a stack of causal convolutional layers and dilated convolutional layers, where each has 3 hidden layers, orange and green notes represent input and output, respectively. (I) A stack of causal convolutional layers with a receptive field of 5. (II) A stack of dilated convolutional layers with a receptive field of 16.

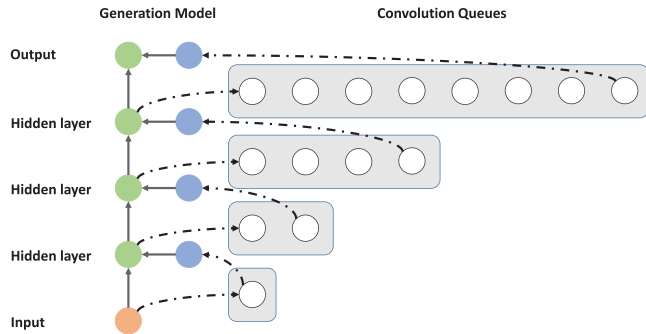


Fig. 4. The diagram of fast dilations. The generation model is initialized with weights from the previous network. After initialization we repeat the pop and push phase iteratively.

convolutions. Because the model has no recurrent connections, the training process is faster than neural networks with recurrent structure. It is important for training long time series. One problem of the casual CNN is that it has a narrow receptive field. Its number of receptive field is similar to the number of layers. To solve it we utilize dilated convolutions to increase the receptive field without significantly increasing computational cost. The key idea of dilated CNN is that it skips the input values in each layer with certain steps, which allows the network to operate on a larger scale that increases the receptive field by orders of the number of layers. As shown in Fig. 3 (II), the dilation is $\{1, 2, 4, 8\}$, while in casual CNN the dilation remains $\{1, 1, 1, 1\}$.

Moreover, to accelerate the computation, we implement dilated CNN in an efficient manner. *Popping* and *pushing* recurrent states in the cache are applied to eliminate redundant convolution operations [26]. It reduces computations from $O(2^L)$ to $O(L)$, where L is the number of layers. Fig. 4 illustrates how it works when dilation is doubled in each layer. Specifically, in the *pop* phase the convolution queue pops the recurrent state and feeds it to the corresponding state. Then the new hidden states and the output are calculated. While in the *push* phase, the algorithm pushes the new hidden state into the convolution queue of the above layer.

After that, a softmax operator is leveraged to impose the non-linearity using a μ -law transformation [19] to produces a

better reconstruction $f(x_t) - \text{sign}(x_t) \frac{1+\mu|x_t|}{\ln(1+\mu)}$, where $|x_t| < 1$ and $\mu = 255$.

F. Postprocessing

The postprocessing consists of skip connections and a series of operations. Please refer to Appendix for details. Postprocessing and dilated CNN form the DNN part of GluNet.

III. RESULTS

A. Dataset and Model

1) *In Silico Data*: The adult and adolescent virtual subjects from the UVA/Padova T1D simulator were employed. We generated 180 days data with 3 meals per day for the 10 virtual adult and 10 virtual adolescent subjects. The generated glucose dataset is a time series with 288 points per day. The intra-day variability is applied in meal size [27] with variability 10%, meal time with $STD = 60$ mins and meal amount mis-estimation $[-0.3, 0.2]$. Furthermore, 30 minutes of exercise with time variability 10% is considered for 2 days per week, and its time varies in the morning, afternoon or evening. Insulin data has several entries per day. The time of insulin entries can be the same as a meal time, or at different time. These settings are variant enough for simulating subjects' daily data under different conditions. Each dataset was randomly divided into two parts: training set and testing set, where each accounted for 50% of the dataset. In the training set, 90%, 10% partition is set as the training/validation strategy.

2) *Clinical Data*: There are two clinical datasets. The first one was obtained from our ABC4D project [21]. T1D diabetic subjects participated and were monitored for 6 consecutive months with Dexcom CGM devices (San Diego, CA). Information on meals, insulin dosages were recorded by the patients themselves through a dedicated App. Similar to the simulated data, each clinical dataset was randomly divided into training set and testing set, where each accounted for around 90 days (50%) of the dataset. The second dataset was provided by the Blood Glucose Level Prediction Challenge, namely the OhioT1DM dataset [22]. During eight weeks' period, six T1D subjects wore CGM devices and insulin pumps and reported their data regularly. A smart-phone app and a fitness band were used to collect daily events. OhioT1DM dataset is different from other datasets,

because its training set and the testing set have been separated by the Challenge, for around 40 days for training and 10 days for testing, respectively.

3) Model Structure and Comparison: In the model framework we use a CNN with 5 layers. For the first three layers, the hidden units are set to 32, while the top two layer has 64 neurons. We use the sliding window of size 16 time-steps to derive one forecast sample for the future glucose with a many-to-one structure. The inputs are glucose, meal, insulin and time stamps. Only these information are used because we want to have a fair comparison that all methods share the same input data. We compare our results with the neural network for predicting glucose algorithm (NNPG) in [28], the latent variable with exogenous input (LVX) statistical method proposed in [29], the auto regression with exogenous input (ARX) method [30], and the support vector regression (SVR) [12] method. 30 and 60 minutes PH are considered.

B. Evaluation Metrics

The efficiency and accuracy of our proposed method will be demonstrated with the root mean square error (RMSE) results in mg/dL and the mean absolute relative difference (MARD) in %, calculated by

$$\begin{aligned} \text{RMSE} &= \sqrt{\frac{1}{N} \sum_{k=1}^N (y_k - \hat{y}_k)^2}, \\ \text{MARD} &= \frac{1}{N} \sum_{k=1}^N \frac{|y_k - \hat{y}_k|}{y_k}, \end{aligned} \quad (5)$$

where y denotes the CGM measurement and \hat{y} denotes the forecasting results, N is the total data numbers. Also a prediction time lag (or called prediction time delay [28]) is computed to measure the time-shift between the actual and predicted signal which gives the highest cross correlation coefficient between them:

$$\tau_{\text{delay}} = \arg \max_k (\hat{y}_k(k|k - PH) \star y(k)). \quad (6)$$

The prediction time lag provides an accurate estimate of the delay (or offset) in predictions [31]. We use these three criteria because they are widely used in evaluating the glucose forecast in the literature.

C. In Silico Results

Table I demonstrates the forecasting performance of the proposed GluNet, NNPG, LVX, SVR, and ARX algorithm, in terms of the corresponding mean values and standard deviations (STD) of RMSE (mg/dL), MARD (%) for the 10 adults from the UVA/Padova T1D simulator. Table II demonstrates the forecasting performance for the 10 adolescents from the UVA/Padova T1D simulator, with a same setting to Table I.

From Table I, we can see that GluNet has the smallest RMSE and MARD. The mean value of RMSE are much improved (GluNet 8.88 ± 0.77 v.s. others best 12.25 ± 1.40 for PH = 30 mins). This verifies GluNet's superiority that it performs well for virtual adult subjects. Also the shortest Time lag shows

that the algorithm can discover the glucose change promptly. Meanwhile, the GluNet method achieves the best performance in the case of PH = 60 mins, in terms of RMSE, MARD and Time lag. Fig. 5 demonstrates that GluNet has less oscillation, especially in the rising period after meal intake. In the rising and the dropping stage, forecasting results of GluNet can follow the CGM samples more closely than those of other methods. It explains why GluNet has a small time lag. If we observe the curves, most proportional curves of real and GluNet curves match each other. The overall time lag is computed from the mean of time when it has the highest cross-correlation over a period of time (e.g. 3 days). If a larger time-shift is made, despite of reducing the time lag near starting points of the rising or dropping stage occasionally, the cross-correlation of the real and GluNet curves will decrease. That is why the computed time lag is within 1 mins for 30 mins PH. Small time lag improves the hypo-/hyper-glycaemia forecasting performance. Moreover, results for the virtual adolescent group are consist with that for the virtual adults. Smallest mean RMSE, MARD, and Time lag values are achieved, despite of the fact that the adolescent glucose level normally is more difficult to model, as shown in Figs. 6 and 7.

D. Clinical Results

Though *In silico* results of GluNet are good, it is important that all methods are tested using clinical datasets in practice. As mentioned, two clinical datasets are examined in this section carefully.

1) ABC4D Datasets: For the 10 adult subjects from the ABC4D project and considering 30 and 60 minutes prediction horizon, Table III demonstrates the forecasting performance in RMSE, MARD for three-month test data, corresponding to GluNet, NNPG, LVX, SVR, and the ARX algorithm. Fig. 8 shows 1.5 day period glucose measurements (in solid black line) and the forecasting results of different methods for the subject 2. The obtained results for clinic data are consistent with that for the *in silico* data, smaller RMSE and MARD values are achieved for all the subjects and all period time (note that the gap between measurements and curves of all methods near meal time are because of the PH offset). GluNet provides steady forecasting and prompt detection of the glucose change. However, generally most values are much worse than the performances for *in silico* data. It is because for clinical data, the data has lots of long missing periods, and the real glucose measurement is more complicated affected by factors besides meal, insulin and time.

2) OhioT1DM Datasets: Table IV presents the 30-minute and 60-minute forecasting performance for six subjects from OhioT1DM datasets in RMSE, MARD, time lag, corresponding to our proposed GluNet, the NNPG, LVX, ARX and the SVR algorithm. The result is leading in the challenge. Fig. 9 shows 1.5 day period glucose measurements (in solid black line) and the prediction results for the subject 570 corresponding to our proposed GluNet method (in solid red line). As comparison, the results of the NNPG (in solid yellow line), LVX (in dash-dotted

TABLE I
PREDICTION PERFORMANCE FOR THE CONSIDERED PREDICTION METHODS EVALUATED ON THE 10-ADULT VIRTUAL SUBJECTS

PH	Methods	RMSE (mg/dL)	MARD (%)	Time lag (mins)	PH	Methods	RMSE (mg/dL)	MARD (%)	Time lag (mins)
30 minutes	GluNet	8.88±0.77	5.32±0.66	0.83±0.40	60 minutes	GluNet	19.90±3.17	10.55±1.40	16.43±4.07
	NNPG	13.55±1.30 [‡]	7.55±1.10 [‡]	10.12±1.67 [‡]		NNPG	24.69±3.19 [‡]	13.39±2.10 [‡]	30.92±5.11 [‡]
	SVR	14.65±1.78 [‡]	7.58±1.05 [‡]	11.55±2.00 [‡]		SVR	25.34±3.71 [‡]	13.16±1.89 [‡]	32.62±4.35 [‡]
	LVX	12.25±1.40 [‡]	7.29±1.02 [‡]	6.54±2.08 [‡]		LVX	22.41±2.89 [‡]	14.06±2.76 [‡]	35.79±2.79 [‡]
	ARX	13.26±1.25 [‡]	7.88±1.24 [‡]	9.63±1.64 [‡]		ARX	25.73±3.41 [‡]	14.71±2.68 [‡]	40.18±2.71 [‡]

* $p \leq 0.05$ [‡] $p \leq 0.01$ [‡] $p \leq 0.005$.

TABLE II
PREDICTION PERFORMANCE FOR THE CONSIDERED PREDICTION METHODS EVALUATED ON THE 10-ADOLESCENT VIRTUAL SUBJECTS

PH	Methods	RMSE (mg/dL)	MARD (%)	Time lag (mins)	PH	Methods	RMSE (mg/dL)	MARD (%)	Time lag (mins)
30 minutes	GluNet	10.73±1.66	5.77±0.88	1.44±0.72	60 minutes	GluNet	22.65±4.71	11.23±1.96	12.81±2.86
	NNPG	16.72±3.18 [‡]	9.25±2.60 [‡]	9.86±1.99 [‡]		NNPG	32.58±7.43 [‡]	16.55±2.83 [‡]	31.69±4.17 [‡]
	SVR	19.72±4.86 [‡]	8.88±1.48 [‡]	12.20±2.15 [‡]		SVR	34.62±8.54 [‡]	16.29±2.49 [‡]	33.80±4.45 [‡]
	LVX	14.99±3.20 [‡]	8.07±1.24 [‡]	7.36±1.94 [‡]		LVX	31.08±7.78 [‡]	15.31±2.41 [‡]	26.48±3.66 [‡]
	ARX	15.24±2.44 [‡]	8.21±1.37 [‡]	8.16±1.31 [‡]		ARX	33.21±6.84 [‡]	18.57±3.00 [‡]	38.79±2.97 [‡]

* $p \leq 0.05$ [‡] $p \leq 0.01$ [‡] $p \leq 0.005$.

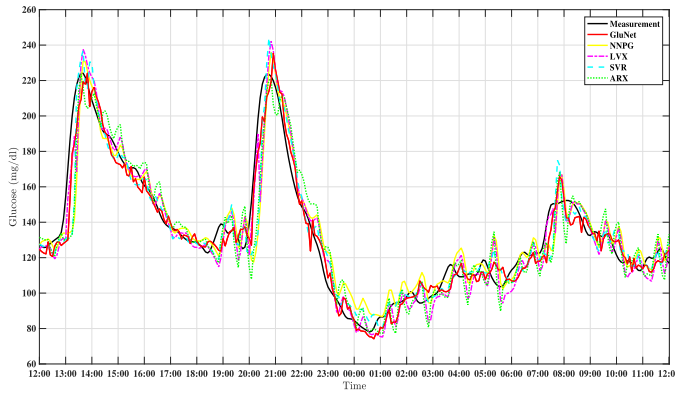


Fig. 5: Virtual adult 2.

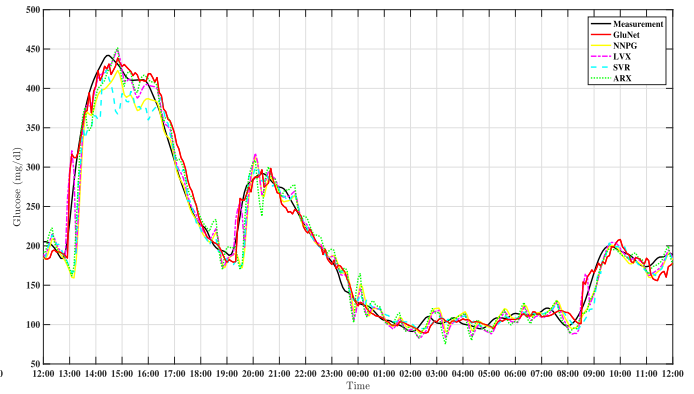


Fig. 6: Virtual adolescent 2.

Fig. 7. One day period forecasting results. The solid black line, solid red line, dash-dotted magenta line, dashed cyan line, and dotted green line indicate the simulated glucose measurements, the forecasting results of the GluNet, NNPG, LVX, SVR, and the ARX method, respectively. (a) Virtual adult 2. (b) Virtual adolescent 2.

TABLE III
FORECASTING PERFORMANCE FOR THE METHODS EVALUATED ON THE 10 ABC4D ADULT SUBJECTS

PH	Methods	RMSE (mg/dL)	MARD (%)	Time lag (mins)	PH	Methods	RMSE (mg/dL)	MARD (%)	Time lag (mins)
30 minutes	GluNet	19.19±2.74	10.41±1.53	11.34±3.09	60 minutes	GluNet	31.78±4.94	19.28±3.85	30.28±6.56
	NNPG	20.42±2.76*	11.91±2.32*	13.88±3.11 [‡]		NNPG	33.13±5.28 [‡]	20.59±4.97*	35.08±4.49*
	SVR	24.96±4.00 [‡]	13.04±2.03 [‡]	14.58±2.93 [‡]		SVR	36.25±5.95 [‡]	21.12±3.77 [‡]	32.36±5.09 [‡]
	LVX	21.52±3.09 [‡]	12.45±1.95*	15.50±3.31 [‡]		LVX	37.34±6.40 [‡]	20.06±3.40	41.44±5.27 [‡]
	ARX	21.24±3.01 [‡]	12.15±2.22*	17.80±2.35 [‡]		ARX	36.17±5.65 [‡]	19.85±3.25	46.29±2.57 [‡]

* $p \leq 0.05$ [‡] $p \leq 0.01$ [‡] $p \leq 0.005$.

magenta line), SVR (in dashed cyan line), and ARX method (in dotted green line) are showed.

It is noted that GluNet still has the best RMSE and MARD performance for the OhioT1DM subjects in Table IV. GluNet largely reduces the overall forecasting error and outperforms other methods. Though GluNet performs well, the improvements are not as large as in the last dataset. The reason might be the short length of training set. In ABC4D dataset, 90 days data are used in the training, while in OhioT1DM dataset only 40 days data are available for training. Longer training data usually

offers some improvements empirically, in particular when we don't have much data at hand. In Fig. 9, we can see that the prediction curve by GluNet is sensitive to the change of the glucose levels and obtains small time lags, especially at the turning points of the uptrend and downtrend. Therefore, the results are compatible with that for the ABC4D dataset.

Moreover, in type 1 diabetes management the inter- and intra-patient variability are so large that it is difficult to use a generalized model for accurate prediction if no data is available from a new subject. However, in some real clinical scenarios we only

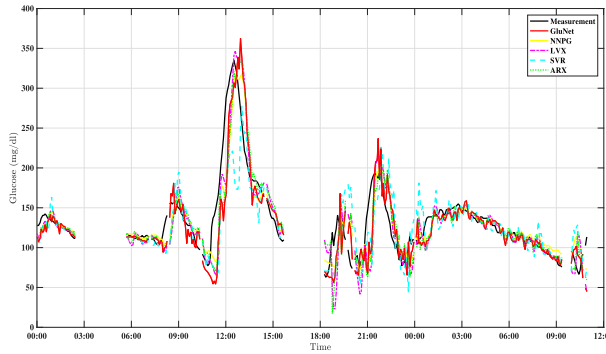


Fig. 8: 1.5 day period forecasting results for adult 2.

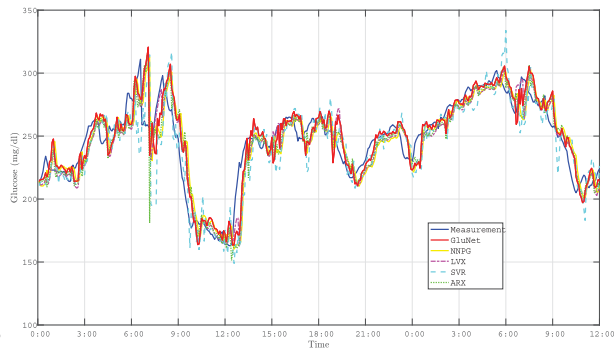


Fig. 9: 1.5 day period prediction results for subject 570.

Fig. 10. 1.5 day period forecasting results for clinical dataset. The solid black line, solid red line, dash-dotted magenta line, dashed cyan line, and dotted green line indicate the simulated glucose measurements, the forecasting results of the GluNet, NNPG, LVX, SVR, and the ARX method, respectively. (a) 1:5 day period forecasting results for adult 2. (b) 1:5 day period prediction results for subject 570.

TABLE IV
FORECASTING PERFORMANCE FOR THE CONSIDERED PREDICTION METHODS EVALUATED ON THE 6 OHIO1DM SUBJECTS

PH	Methods	RMSE (mg/dL)	MARD (%)	Time lag (mins)	PH	Methods	RMSE (mg/dL)	MARD (%)	Time lag (mins)
30 minutes	GluNet	19.28±2.76	8.73±1.78	8.03±4.07	60 minutes	GluNet	31.83±3.49	16.11±4.49	17.78±8.00
	NNPG	22.93±2.95 [‡]	10.76±2.84 [‡]	16.42±4.21 [‡]		NNPG	35.77±3.50 [†]	17.75±4.78*	30.72±15.28 [‡]
	SVR	21.75±1.86*	9.23±2.15	9.86±4.46		SVR	34.31±2.85*	17.86±4.63*	20.89±11.40*
	LVX	21.70±3.28 [‡]	9.50±2.13*	11.75±5.64*		LVX	38.95±12.76 [‡]	17.69±3.17	27.06±12.23 [‡]
	ARX	21.54±2.42 [‡]	9.17±2.06	11.06±6.21*		ARX	39.53±14.47 [‡]	17.58±2.98	35.53±9.38 [‡]

* $p \leq 0.05$ [†] $p \leq 0.01$ [‡] $p \leq 0.005$.

TABLE V
COMPUTATIONAL TIME OF THE METHODS

Method	Inference (s)	Training (s)	Structure Overview
GluNet	0.79	878.11/13.26(GPU)	14592 params, 5 layers
NNPG	0.19	407.19	2196 params, 3 layers
SVR	0.54	127.25	RBF Kernel
LVX	0.05	1.73	3 PLS LVs, CCA [38]
ARX	0.14	4.90	3 order

have limited historical datasets from several recorded subjects and little information of a new subject. To investigate this problem, we evaluated a generalized GluNet model on new subjects. Particularly, we trained the model on data corresponding to five subjects of ABC4D datasets. Then directly tested it on the rest of subjects one by one. The result turns out that the mean RMSE of generalized model is 24.93 mg/dl while the personalized model achieves 20.59 mg/dl, when PH = 30 mins. In addition, if limited new data is available, transfer learning can be used to upgrade the generalized GluNet model to a personalized model, like the work [15] and our work [32]. Thus, GluNet can provide useful prediction to new users without much data, and accuracy can be improved as new individual data becomes available. Integrating with cloud, GPUs and mobiles, it provides us with a convenient pattern to train and update models in practice.

IV. DISCUSSION

A. Advantages and Limitations of GluNet

From the results, we can see that GluNet has several advantages. Firstly, it has a higher forecasting accuracy compared

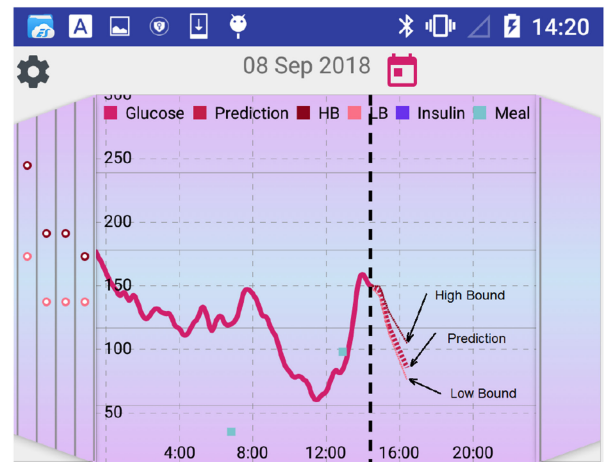


Fig. 11. The display for high and low bounds with the corresponding PH.

to other existing algorithms (also see the comparison details in the next subsection). Secondly, it is simple to implement, without tedious parameter tuning. Thirdly, it provides richer information of the future glucose level by means of a probabilistic distribution rather than a single value forecasting. Most importantly, GluNet has the shortest time lag and can capture the trend promptly. Here, we implement the proposed algorithm in an Android system on smart phones using TensorLite. The app offers the glucose forecasting as shown in Fig. 11, as well as its lower and higher bounds with 95% probability according to the probabilistic distribution. It uses around 50 MB RAM when

sleeping and 100 MB RAM to make the forecasting. The forecasting only takes around 6 ms in the model inference, which is much faster than running it in Python on a laptop.

One potential limitation of GluNet is that the algorithm heavily relies on the training data quality. In simulation, the data has longer length and has less variability. Thus when we have 6 months high quality data as in the simulation, the forecasting accuracy and time lag can be very good. However, if long data missing period exists, such as in the ABC4D dataset, or the training data is less than two months (in OhioT1DM) with high variability, the performance is significantly affected. In future work, the physiological model could be considered in the DNN model to improve the forecasting performance under these cases.

B. Comparison With Other Existing Algorithms

We compare GluNet with existing algorithms to show its effectiveness, including NNPG, SVR, LVX, and ARX. Other algorithms published in literature have also been considered. For datasets generated from the simulator, our proposed method is better than [33] (RMSE = 18.8), [34] (RMSE = 13.7). For real data, GluNet's RMSE 19.2 is better than 23.4 in [35], and 21.8 in [36]. GluNet's time lag 11.3 mins is much better than 20.4 mins in [36] for 30 mins PH. However, due to the unavailability of the original codes and other paper's RMSE, it is not easy to have a fair comparison. We do however try to compare our results with works that have been recently published in the following section.

In [37] the algorithm was compared to ARMA, SVR and doctors on clinical data in terms of RMSE with PH = 30 mins. Because we do not have their clinical data, comparing the RMSE directly is not fair. Instead we found that their proposed algorithm is 0.1 better than SVR in terms of RMSE. While GluNet gains 5.7 better than SVR on our clinical data, which implies its advantage. The GluNet's RMSE result of 8.88 for virtual adults is also better than the RMSE of 9.40 achieved in [10]. GluNet's result was run on 90 training dates and 90 testing dates, which is more convincing than 11 dates in training and 3 dates in testing from which the result was obtained in [10]. To summarise, GluNet achieves the state-of-the-art forecasting accuracy with the shortest time lag when compared with published methods.

C. Parameter Tuning and Training

For control-theory based algorithms, usually many parameters need to be tuned manually. The forecasting accuracy can be influenced significantly if non-ideal parameters are adopted. In GluNet, several hyper-parameters are set before the training, and these values are fixed for all subjects, such as sliding window size, learning rate followed by the network size. In fact, default learning rate is applied in GluNet. While values of most other parameters are automatically derived via training. At the end of the training, the validation loss is quite close to the training loss, and both loss values are converging. It implies that our model capacity is appropriate and it shows good generalization of the model. Early stopping technique is used and it prevents

the overfitting before running too many epochs. Unlike other glucose forecasting algorithms, GluNet offers a probabilistic distribution of the forecasting glucose over the next PH period. Thus it provides richer information for the users regarding the future glucose trend.

D. Computational Cost

To compare computational cost of GluNet with other methods, we run the models on the subjects from the OhioT1DM datasets using the same hardware. Table V presents the computational time of 10 simulations that train the model on 40-day historical data and then obtain the inference of 10-day period. For real-time applications, the inference time is important. Although GluNet consumes more time (878.11 s) than other models, all five models have good performance on inference mode (less than one second). Compared with many conventional CNN applications, GluNet has a light structure, yet it still has much more parameters than other considered models. In real-time implementation of such a system, we will train the model on the cloud using GPUs and download the trained model into mobile devices to execute prediction tasks. In this case, it only takes 6 ms on an Android system to obtain a single prediction for next 30 minutes, and 13.26 s on Google Cloud GPU to train the same amount of data.

V. CONCLUSION AND FUTURE WORK

In this paper, a glucose forecasting approach called GluNet was proposed using deep neural networks. GluNet achieved better glucose forecasting results in terms of accuracy (RMSE and MARD) and in particular time lag. The proposed approach provides an personalized predictive model by using individual data, and algorithm's computation cost is light enough to be implemented on a mobile phone in real time. GluNet was tested intensively on *in silico* virtual adult, virtual adolescent and clinical adult datasets with leading performance.

In the future, we consider how to incorporate physiological knowledge in our model. Currently we use them as optional inputs to further improve the its performance. A better approach is in investigation. We also plan to evaluate GluNet with longer prediction horizons and apply it for over-night hypoglycemia prediction.

APPENDIX

A. Postprocessing

If \mathbf{W} denotes the learnable convolution filter,

$$\mathbf{y} = \tanh(\mathbf{W}_{f,k} \star \mathbf{x}) \odot \sigma(\mathbf{W}_{g,k} \star \mathbf{x}), \quad (7)$$

where \mathbf{z} is the output of unit, \star , \odot are the convolution operator and element-wise multiplication operator, respectively. \tanh denotes the tangent function and σ denotes the sigmoid function. f, g, k denote filter, gate and layer index. We use the residual in [39] and skip connections throughout the work.

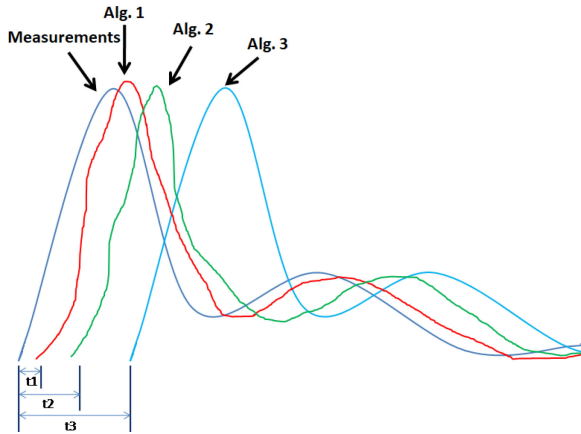


Fig. 12. Graphical representation of the ‘prediction time lag’. Consider Alg. 1, Alg. 2 and Alg. 3 (Alg. 3 is to output the current CGM measurements as the prediction result), and the PH is set to 30 minutes. Then, the time lags of Alg. 1,2,3 are approximately t_1 , t_2 and t_3 , respectively, and $t_3 = 30$ minutes.

B. Explanation of Time Lag

We use Equation (6) to compute the ‘prediction time lag’ [31], [40]. This definition is different from the ‘CGM time lag’ that denotes the time lag between the blood glucose and CGM measurements. Fig. 12 shows a diagram of the time lag. In practice it is not easy to obtain the time lag directly, hence the cross correlation is used to compute it. From the example one can see that a shorter time lag generally implies a better prediction, because the algorithm with a shorter time lag captures the curve changes quicker than other algorithms.

REFERENCES

- [1] K. Ogurtsova *et al.*, “IDF diabetes atlas: Global estimates for the prevalence of diabetes for 2015 and 2040,” *Diabetes Res. Clin. Pract.*, vol. 128, pp. 40–50, 2017.
- [2] F. Barcelo-Rico, J. Diez, P. Rossetti, J. Vehi, and J. Bondia, “Adaptive calibration algorithm for plasma glucose estimation in continuous glucose monitoring,” *IEEE J. Biomed. Health Informat.*, vol. 17, no. 3, pp. 530–538, May 2013.
- [3] A. Facchinetti, “Continuous glucose monitoring sensors: Past, present and future algorithmic challenges,” *Sensors*, vol. 16, no. 12, p. 2039, 2016.
- [4] T. Koutny and M. Ubl, “Parallel software architecture for the next generation of glucose monitoring,” *Procedia Comput. Sci.*, vol. 141, pp. 279–286, 2018.
- [5] M. Vettoretti, A. Facchinetti, G. Sparacino, and C. Cobelli, “Type 1 diabetes patient decision simulator for in silico testing safety and effectiveness of insulin treatments,” *IEEE Trans. Biomed. Eng.*, vol. 65, no. 6, pp. 1281–1290, Jun. 2018.
- [6] P. Pesi *et al.*, “An advanced bolus calculator for type 1 diabetes: System architecture and usability results,” *IEEE J. Biomed. Health Informat.*, vol. 20, no. 1, pp. 11–17, Jan. 2016.
- [7] K. Li, F. Liu, H. Dong, P. Herrero, Y. Guo, and P. Georgiou, “A deep learning platform for diabetes big data analysis,” in *Proc. 11th Adv. Technologies Treatments Diabetes*, 2018, pp. A116–A116.
- [8] S. Oviedo, J. Vehi, R. Calm, and J. Armengol, “A review of personalized blood glucose prediction strategies for T1DM patients,” *Int. J. Numer. Methods Biomed. Eng.*, vol. 33, no. 6, 2017, Art. no. e2833.
- [9] M. Eren-Oruklu, A. Cinar, L. Quinn, and D. Smith, “Estimation of future glucose concentrations with subject-specific recursive linear models,” *Diabetes Tech. Therapeutics*, vol. 11, no. 4, pp. 243–253, Apr. 2009.
- [10] C. Zecchin, A. Facchinetti, G. Sparacino, G. D. Nicolao, and C. Cobelli, “Neural network incorporating meal information improves accuracy of short-time prediction of glucose concentration,” *IEEE Trans. Biomed. Eng.*, vol. 59, no. 6, pp. 1550–1560, Jun. 2012.
- [11] C. Zhao *et al.*, “Predicting subcutaneous glucose concentration using a latent-variable-based statistical method for type 1 diabetes mellitus,” *J. Diabetes Sci. Technol.*, vol. 6, no. 3, pp. 617–633, 2012.
- [12] E. I. Georga *et al.*, “Multivariate prediction of subcutaneous glucose concentration in type 1 diabetes patients based on support vector regression,” *IEEE J. Biomed. Health Informat.*, vol. 17, no. 1, pp. 71–81, Jan. 2013.
- [13] J. Li and C. Fernando, “Smartphone-based personalized blood glucose prediction,” *ICT Express*, vol. 2, no. 4, pp. 150–154, 2016, special Issue on Emerging Technologies for Medical Diagnostics. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S2405959516301126>
- [14] I. Contreras, S. Oviedo, M. Vettoretti, R. Visentin, and J. Vehi, “Personalized blood glucose prediction: A hybrid approach using grammatical evolution and physiological models,” *Plos One*, vol. 12, no. 11, pp. 1–16, Nov. 2017. [Online]. Available: <https://doi.org/10.1371/journal.pone.0187754>
- [15] N. M. Rad *et al.*, “Deep learning for automatic stereotypical motor movement detection using wearable sensors in autism spectrum disorders,” *Signal Process.*, vol. 144, pp. 180–191, 2018.
- [16] K. Zarkogianni *et al.*, “Comparative assessment of glucose prediction models for patients with type 1 diabetes mellitus applying sensors for glucose and physical activity monitoring,” *Med. Biol. Eng. Comput.*, vol. 53, no. 12, pp. 1333–1343, Dec. 2015.
- [17] H. N. Mhaskar, S. V. Pereverzyev, and M. D. van der Walt, “A deep learning approach to diabetic blood glucose prediction,” *Frontiers Appl. Math. Statist.*, vol. 3, p. 14, 2017. [Online]. Available: <https://www.frontiersin.org/article/10.3389/fams.2017.00014>
- [18] A. van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, and A. Graves, “Conditional image generation with pixelcnn decoders,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4790–4798.
- [19] A. van den Oord *et al.*, “Wavenet: A generative model for raw audio,” 2016. [Online]. Available: <https://arxiv.org/pdf/1609.03499>
- [20] C. D. Man, F. Micheletto, D. Lv, M. Breton, B. Kovatchev, and C. Cobelli, “The uva/padova type 1 diabetes simulator,” *J. Diabetes Sci. Technol.*, vol. 8, no. 1, pp. 26–34, Jan. 2014.
- [21] M. Reddy *et al.*, “Clinical safety and feasibility of the advanced bolus calculator for type 1 diabetes based on case-based reasoning: A 6-week nonrandomized single-arm pilot study,” *Diabetes Technol. Therapeutics*, vol. 18, no. 8, pp. 487–493, 2016. [Online]. Available: <https://doi.org/10.1089/dia.2015.0413>
- [22] C. Marling and R. Bunescu, “The OhioT1DM dataset for blood glucose level prediction,” in *Proc. 3rd Int. Workshop Knowl. Discovery Healthcare Data*, Stockholm, Sweden, Jul. 2018. [Online]. Available: <http://smarthealth.cs.ohio.edu/bglp/OhioT1DM-dataset-paper.pdf>
- [23] P. Herrero *et al.*, “A simple robust method for estimating the glucose rate of appearance from mixed meals,” *J. Diabetes Sci. Technol.*, vol. 6, no. 1, pp. 153–162, 2012.
- [24] T. Koutny, “Glucose-level interpolation for determining glucose distribution delay,” in *Proc. 13th Mediterranean Conf. Med. Biol. Eng. Comput.*, 2014, pp. 1229–1232.
- [25] R. Hovorka *et al.*, “Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes,” *Physiological Meas.*, vol. 25, no. 4, pp. 905–920, 2004.
- [26] T. L. Paine *et al.*, “Fast wavenet generation algorithm,” 2016, *arXiv:1611.09482*.
- [27] C. Dalla Man, R. A. Rizza, and C. Cobelli, “Meal simulation model of the glucose-insulin system,” *IEEE Trans. Biomed. Eng.*, vol. 54, no. 10, pp. 1740–1749, Oct. 2007.
- [28] C. Pérez-Gandía *et al.*, “Artificial neural network algorithm for online glucose prediction from continuous glucose monitoring,” *Diabetes Tech. Therapeutics*, vol. 12, no. 1, pp. 81–88, 2010.
- [29] C. Zhao, E. Dassau, L. Jovanović, H. C. Zisser, F. J. Doyle III, and D. E. Seborg, “Predicting subcutaneous glucose concentration using a latent-variable-based statistical method for type 1 diabetes mellitus,” *J. Diabetes Sci. Technol.*, vol. 6, no. 3, pp. 617–633, 2012.
- [30] D. A. Finan *et al.*, “Experimental evaluation of a recursive model identification technique for type 1 diabetes,” *J. Diabetes Sci. Technol.*, vol. 3, no. 5, pp. 1192–1202, 2009.
- [31] A. Gani, A. V. Gribok, Y. Lu, W. K. Ward, R. A. Vigersky, and J. Reifman, “Universal glucose models for predicting subcutaneous glucose concentration in humans,” *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 1, pp. 157–165, Jan. 2009.

- [32] J. Chen, K. Li, P. Herrero, T. Zhu, and P. Georgiou, "Dilated recurrent neural network for short-time prediction of glucose concentration," in *Proc. 3rd Int. Workshop Knowl. Discovery Healthcare Data*, Stockholm, Sweden, Jul. 13, 2018, pp. 69–73.
- [33] G. Sparacino, F. Zanderigo, S. Corazza, A. Maran, A. Facchinetti, and C. Cobelli, "Glucose concentration can be predicted ahead in time from continuous glucose monitoring sensor time-series," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 5, pp. 931–937, May 2007.
- [34] S. G. Mougiakakou, A. Prountzou, D. Iliopoulou, K. S. Nikita, A. Vazeou, and C. S. Bartsocas, "Neural network based glucose—Insulin metabolism models for children with type 1 diabetes," in *Proc. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2006, pp. 3545–3548.
- [35] A. Gani, A. V. Gribok, S. Rajaraman, W. K. Ward, and J. Reifman, "Predicting subcutaneous glucose concentration in humans: Data-driven glucose modeling," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 2, pp. 246–254, Feb. 2009.
- [36] Q. Sun, M. V. Jankovic, L. Bally, and S. G. Mougiakakou, "Predicting blood glucose with an LSTM and Bi-LSTM based deep neural network," in *Proc. 14th Symp. Neural Net. Appl.*, Nov. 2018, pp. 1–5.
- [37] K. Plis, R. Bunescu, C. Marling, J. Shubrook, and F. Schwartz, "A machine learning approach to predicting blood glucose levels for diabetes management," in *Proc. Modern Artif. Intell. Health Anal. Papers*, 2014, pp. 35–39.
- [38] H. Yu and J. F. MacGregor, "Post processing methods (PLS–CCA): Simple alternatives to preprocessing methods (OSC–PLS)," *Chemometrics Intell. Lab. Syst.*, vol. 73, no. 2, pp. 199–205, 2004.
- [39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [40] E. I. Georga, V. C. Protopappas, and D. I. Fotiadis, "Glucose prediction in type 1 and type 2 diabetic patients using data driven techniques," *Knowl. Oriented Appl. Data Mining*, 2011, pp. 277–296.