Personalized LSTM-based alarm systems for hypoglycemia and hyperglycemia prevention

Francesca Iacono[a], Lalo Magni[b], Chiara Toffanin[a,*]

[a] *Department of Electrical, Computer and Biomedical Engineering, University of Pavia, via Ferrata 3, Pavia, 27100, Italy*
[b] *Department of Civil and Architecture Engineering, University of Pavia, via Ferrata 3, Pavia, 27100, Italy*

Abstract

Hypoglycemia and hyperglycemia prevention is the main challenge of an efficient Type 1 Diabetes (T1D) control. Alarm systems that alert the patients when their Blood Glucose (BG) levels are going to be critical can be useful instruments in order to react and avoid upcoming hypoglycemia and hyperglycemia events. These alarm systems can be used with both the conventional basal-bolus therapy or in conjunction with the advanced closed-loop control system, the so-called artificial pancreas. Model-based alarms use patient models to predict future BG levels and then activate alarms, so these models have to be reliable and to ensure good performances. In recent studies, neural network techniques for glucose forecasting obtained promising results, for both population and personalized models. These recent works showed that personalized Long Short-Term Memory (LSTM) models for BG predictions obtained good results on the 100 in silico patients of the most recent version of the UVA/Padova simulator. In this work personalized alarm systems for hypoglycemia and hyperglycemia prediction based on personalized LSTM models are proposed. Promising results have been obtained, detecting correctly the 77% of the hypoglycemia and the 89% of the hyperglycemia events.

Keywords: Diabetes, alarm systems, hyperglycemia/hypoglycemia avoidance,

*Corresponding author
Email addresses: francesca.iacono01@universitadipavia.it (Francesca Iacono), lalo.magni@unipv.it (Lalo Magni), chiara.toffanin@unipv.it (Chiara Toffanin)

neural networks, personalized models, glucose prediction.

1. Introduction

Diabetes refers to a group of common metabolic conditions that share the phenotype of hyperglycemia, i.e. when Blood Glucose (BG) reaches high values (BG>180 mg/dl). Type 1 Diabetes (T1D) is the result of complete or near-total insulin deficiency due to an autoimmune process against the β-pancreatic cells, cells that produce the insulin. Indeed, T1D is defined an insulin-dependent disease, i.e. it can only be treated with exogenous insulin injections to decrease the BG levels, that involve risks of hypoglycemia (BG<70 mg/dl) if excessive. In the next years T1D incidence in the population is supposed to largely increase. The International Diabetes Federation reports approximately 537 million adults with diabetes and an expected increase to 643 million of people living with diabetes by 2030 [1]. Then, the T1D optimal treatment is a challenging and interesting control problem.

The T1D conventional treatment is the so-called Basal-Bolus Therapy (BBT) constituted of a piecewise constant amount of insulin delivered along the day during fasting periods, called basal, and some meal boluses. A meal bolus is an impulse-like amount of insulin delivered to compensate the glucose rise due to a meal intake, computed using the estimation of the carbohydrate intake of the meal [2]. Along with the conventional therapy, in the past years the use of Continuous Glucose Monitoring (CGM) sensors and insulin pumps spread around in the so-called Sensor-Augmented Pump (SAP) system, since sensors became more precise and reliable. The use of SAP systems brought an improvement in the glucose control, reducing the incidence of adverse episodes and increasing at the same time treatment satisfaction and diabetes related distress [3–5]. Moreover, at the same time systems called Artificial Pancreas (APs) have been developed to optimally administrate insulin, in order to prevent both hypoglycemia and hyperglycemia phenomena in T1D patients. An AP is a closed-loop system that computes and directly delivers the optimal insulin quantity through

an insulin pump, exploiting CGM measurements and other additional information provided by the patient to estimate this optimal quantity. Thanks to the creation of the UVA/Padova simulator [6], a metabolic simulator approved by Food and Drug Administration (FDA) as a substitute to animal trials, the AP's development had a notable acceleration. The core of the AP is the control algorithm devoted to the estimation of the optimal insulin therapy. One of the most promising algorithm, widely used in this kind of application, is the Model Predictive Control (MPC) [7–13]. This algorithm forecasts future BG levels exploiting a patient model in order to estimate the optimal quantity of insulin that keeps BG within the safe range ([70-180] mg/dl).

Since hypoglycemia and hyperglycemia prevention is one of the key points of an effective diabetes management, to preserve the patients safety, CGM devices, SAP systems and AP can be equipped with Alarm Systems (ASs) that alert the patient of an upcoming critical event. Depending on the method that generates the alert, the ASs can be categorized into two types: threshold detection and prediction-based. Threshold detection alarms are activated when particular thresholds are crossed, in this case when critical BG levels are reached [14], while prediction-based alarms attempt to evaluate the risk beforehand, relying on patient models, which goodness determines the AS performance.

In this latter application, models for glucose prediction assume a huge relevance. Thanks to a growing availability of both in silico and in vivo data [15], different neural network techniques have been applied in this field in the last years. For example multilayer perceptron [16], reinforcement learning techniques [17] and above all the deep learning models [18] were object of recent study. Among the different deep learning architectures, Recurrent Neural Networks (RNNs) are particularly performing for this task, since they are specialized in the analysis of sequential values. In particular, a type of the RNNs, the Long Short-Term Memory (LSTM) network [19, 20], showed promising results [21–26]. In [21] and [22] only past CGM data are used to predict future BG values, since CGM data are more easily obtainable only wearing a sensor, with respect to meals and insulin information. In fact, insulin and meal information requires to be

recorded manually by the patients or automatically by devices like APs. However, a significant improvement can be obtained using also them as inputs if the Prediction Horizon (PH) is greater than 30 minutes, as studied in [27]. These additional features were employed to train LSTMs with more complex architectures for population models obtaining interesting results in [23–26].

Considering the inter- and intra-patient variability that characterizes the BG prediction problem, i.e. each patient is different from another and their behaviour changes over the time, population and time-invariant models can demonstrate limited performances in glucose prediction. In fact, personalized and time-variant models represent a promising choice for this type of applications. In a previous work [28], Personalized LSTM models (P-LSTMs) developed for the 100 in silico patients of the UVA/Padova simulator [6] obtained interesting results. The P-LSTMs predicted future CGM values with a PH of 40 [min], having in input past CGM values, current meals and current insulin quantities. The obtained results compared with the literature [21–26], showed that a simple network architecture is able to accurately predict future BG levels, taking into account the inter-patient variability. Starting from the results obtained in [28], in this work the P-LSTMs are improved to achieve better performance and meet some accuracy requirements: a set of Enhanced Personalized LSTM models (EP-LSTMs) is proposed as first result of this work. Then, the EP-LSTMs are used to design personalized ASs for hypoglycemia and hyperglycemia detection.

This paper is divided in 4 main sections: Section 2, devoted to the EP-LSTM design, describes the LSTM architecture and the datasets used for the neural network application, together with the state of the art; Section 3 presents the AS developed exploiting EP-LSTM, describing how the alarm system works and the dataset used for its evaluation; Section 4 reports the results and in Section 5 the conclusions are drawn.

2. Enhanced Personalized LSTM

2.1. LSTM for glucose prediction

A Long Short-Term Memory (LSTM) network, firstly introduced by [19, 20], is a particular type of RNN [29]. The structure of a RNN is characterized by a recurrent loop where the output of a node is fed-back in input to the replica of the node at the next time instant, propagating the long-term dependencies through time. It is well known that the RNNs suffer during the training phase of the vanishing gradient problem, so learning temporal dependencies becomes challenging [30]. In fact, the training is based on the backpropagation algorithm [31] during which the gradient may vanish or explode, leading in a severe worsening of the optimization [32]. To solve this issue, some particular architectures have been designed, like the LSTM networks that thanks to their specific units architecture, called memory cells, are able to learn the information dependencies better than the simple RNNs. The memory cells are designed with an internal recurrence, in addition to the outer self-loop of the RNNs, and a gating system that regulates the flow of information. In this way, the gradient flows during time, even for long periods, and its derivatives do not explode nor vanish.
The mathematical formulation of a single memory cell (see Figure 1) is expressed as follows:

$$c_k = f\left(x_k, h_{k-1}\right) \times c_{k-1} + i\left(x_k, h_{k-1}\right) \times \tilde{c}\left(x_k, h_{k-1}\right) \tag{1a}$$

$$h_k = o\left(x_k, h_{k-1}\right) \times tanh\left(c_k\right) \tag{1b}$$

where $c_k \in \mathbb{R}^{n_c}$ is the internal cell state at time k, $x_k \in \mathbb{R}^{n_x}$ is the input at time k and $h_k \in \mathbb{R}^{n_c}$ is the hidden state at time k, with n_c the cell state dimension, that corresponds to the number of LSTM neurons, n_x the number of features, $\times$ is the Hadamard product and $tanh$ is the hyperbolic tangent, used as activation function.
Four structures, called gates, compose the internal loop of the cell and add or remove information to modify the value of the cell state. The first gate, $f\left(x_k, h_{k-1}\right)$, is the forget gate that decides which information is removed from

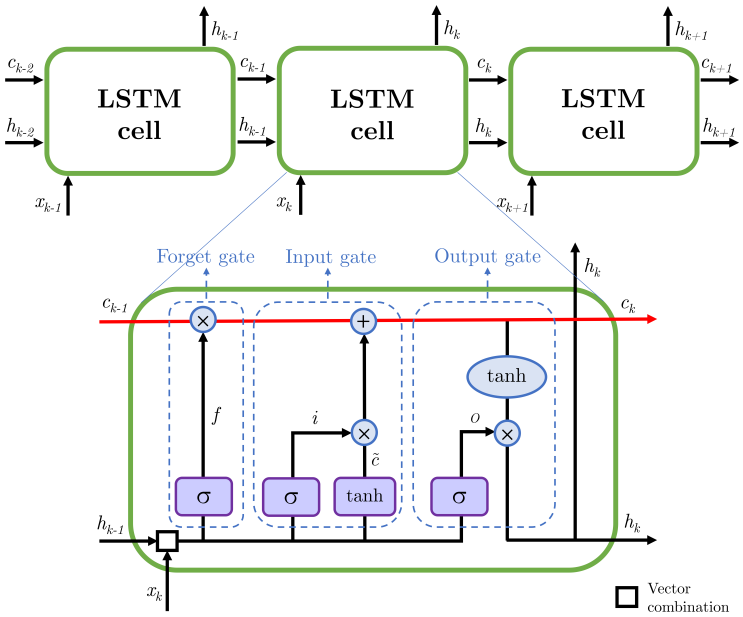


Figure 1: Scheme of a single LSTM cell.

the cell state c_k, multiplying it by its past value c_{k-1}, as can be seen in the first part of (1a). Then the input gate, $i\left(x_k, h_{k-1}\right)$, chooses which value of the cell state is updated, while the input activation gate, $\tilde{c}\left(x_k, h_{k-1}\right)$, creates a new candidate value of the cell state. The last gate is the output gate, $o\left(x_k, h_{k-1}\right)$, that determines which information contained in the cell state is going in output. The cell output is defined in (1b), where the output gate is multiplied by a transformation through hyperbolic tangent function of the cell state. These gates are expressed as:

$$
\begin{aligned}
f\left(x_k, h_{k-1}\right) &= \sigma(W_f x_k + U_f h_{k-1} + b_f) \\
i\left(x_k, h_{k-1}\right) &= \sigma(W_i x_k + U_i h_{k-1} + b_i) \\
\tilde{c}\left(x_k, h_{k-1}\right) &= tanh(W_c x_k + U_c h_{k-1} + b_c) \\
o\left(x_k, h_{k-1}\right) &= \sigma(W_o x_k + U_o h_{k-1} + b_o)
\end{aligned}
$$

where $W_f, W_i, W_o, W_c \in \mathbb{R}^{n_c \times n_x}$ are the input weight matrices, $U_f, U_i, U_o, U_c \in \mathbb{R}^{n_c \times n_c}$ are the previous output weight matrices and $b_f, b_i, b_o, b_c \in \mathbb{R}^{n_c \times 1}$ are

the biases. These quantities are specific of each gate and obtained during the training of the network. The function σ is the sigmoid function:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

A single LSTM layer is composed by a variable number of neurons (n_c) and the network can be composed by one or more layers. The output of the last LSTM layer is passed to a fully connected layer, used as output layer of the network and whose formulation is:

$$y_k = W_y h_k + b_y$$

where $W_y \in \mathbb{R}^{n_p \times n_c}$ is the matrix of the output weights, with n_p the number of outputs of the network, and b_y the bias, a scalar number. W_y and b_y are determined during the training of the network as well.

In this work a network composed by a single layer LSTM with a fully connected layer, with three inputs ($n_x = 3$) and one output ($n_p = 1$) is proposed; the number of neurons n_c is one of the LSTM parameters used for personalization. The inputs are the injected insulin through an insulin pump, $I(k)$, the meals intake provided by the patient, $M(k)$, and the past CGM values of PH minutes before, $CGM(k - PH)$; the output is the CGM value at the current time, $CGM(k)$.

2.2. Data generation

In this work three different datasets are used to train and evaluate the LSTM: a training dataset (*tr-dataset*), used to train the network; a validation dataset (*v-dataset*) for the hyperparameters tuning; a testing dataset (*ts-dataset*), used to assess the network's performance. The UVA/Padova simulator [6] have been used to generate the in silico data. This simulator has been approved by the FDA and used to test insulin therapies in silico, as an alternative to animal experiments, being demonstrated that its simulations reflect the glucose patterns observed in human studies on T1D patients [33]. The simulator provides a virtual population composed by 300 subjects, with 100 adults, 100 adolescents and 100 children, ensuring to represent the inter-patient variability of the real T1D

population. Moreover, the most recent version of the simulator [6] includes also the intra-patient variability, representing glucose variations of insulin sensitivity during the day and allowing a realistic simulation. A complete description of the UVA/Padova simulator is provided in Appendix A. In this work the 100 adult subjects of this most advanced version of the simulator are considered.

2.2.1. In silico datasets

The entire adult population of the UVA/Padova simulator is used to generate the datasets that include CGM, insulin and meals; the insulin quantity is computed by the MPC described in [34]. Since the goal of this work is the detection of hypoglycemia and hyperglycemia events due to a not-optimal treatment, a significant amount of critical episodes in all the datasets is needed, so the MPC is tuned in order to be sub-optimal. In this way, the insulin therapy defined by the MPC is closer to other possible treatments defined by the physician that result not-optimal in presence of uncertainties. Moreover, it is possible to reproduce the glucose dynamic of patients with regulation problems that will mainly benefit of a personalized model and of a safety system. A complete description of the sub-optimal MPC control strategy is reported in Appendix A. It is important to note that the insulin profiles can be generated with any other control algorithm (e.g. using the BBT) in order to mimic the patient's insulin therapy and his/her glucose regulation in real-life conditions. The three scenarios presented in [28] are employed also in this work to generate the *tr-dataset*, *v-dataset* and *ts-dataset*. All the datasets include three meals per day and up to two additional snacks, in different quantities and times of the day. The number of meals per day and the time distribution are inspired by the ones observed on the real patients of the Padova center [35] in order to reproduce realistic conditions, since random distributions could not represent real-life scenarios. However, to ensure uncorrelation between the datasets, there are no repetitions in the meal characteristics of each day that composes them. A complete description of the meal distribution is presented in Table 1: times and quantities of the meals in each dataset are reported in terms of mean ($\pm$ SD)

Dataset	Meals	CHO (g)	Time
	Breakfast	59 ± 22	7:08 ± 81
tr-dataset	Lunch	63 ± 11	12:45 ± 60
(8 days of data)	Dinner	60 ± 17	20:26 ± 56
	Snacks	25 ± 7	
	Breakfast	50 [50-52.5]	7:30 ± 24
v-dataset	Lunch	66 ± 9	13:22 ± 29
(4 days of data)	Dinner	72 ± 6	20:15 ± 52
	Snacks	23 ± 7	
	Breakfast	57 ± 6	7:22 ± 75
ts-dataset	Lunch	65 ± 4	13:15 ± 39
(4 days of data)	Dinner	86 ± 12	19:52 ± 29
	Snacks	19 ± 4	

Table 1: Datasets descriptions

or median [25^{th} - 75^{th}] percentiles, if the data are normally or not normally distributed, respectively. Note that, since the snacks can be assumed throughout the day, only their quantities are reported, seen their time distribution is not significant.

In view of this meal variability, the *tr-dataset*, *v-dataset* and *ts-dataset* present significant differences both from one day to another and between the datasets as shown in Figure 2. In the figure the median [25^{th} - 75^{th}] percentiles of the glucose profiles of the entire population are reported for all the 3 datasets, since they are not normally distributed. To provide more realistic settings during the data generation phase, the meal announcement presents inaccuracies in terms of amount (CHO) and time of the carbohydrate intake. In Figure 3 the distributions of the bolus delays and the CHO counting errors introduced in all the datasets are reported for each type of meal. The scenarios used to generate the *tr-dataset*, *v-dataset* and *ts-dataset* are presented in detail in Appendix A.

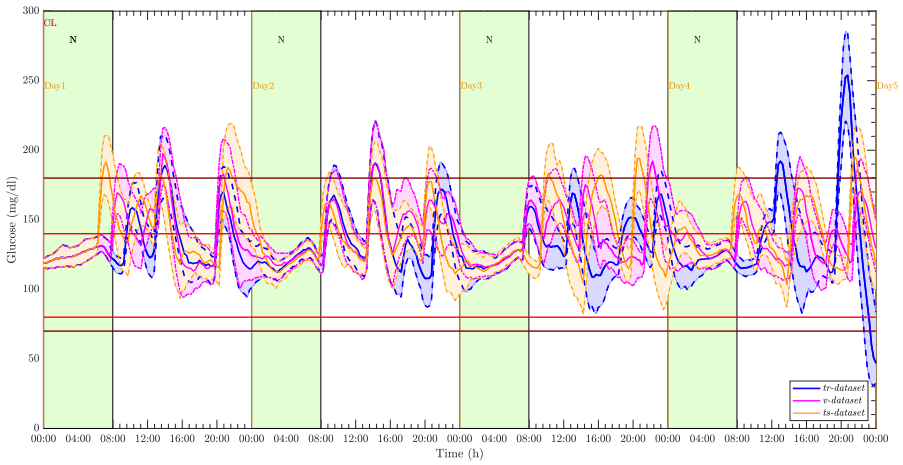


Figure 2: Blood glucose profiles of 100 in silico patients with *tr-dataset* (blue), *v-dataset* (magenta) and *ts-dataset* (orange). Glucose profiles are shown in terms of median (solid lines) surrounded by colored regions representing the [25^{th} - 75^{th}] percentiles. N, night period. Euglycemic range (80-140 [mg/dl]) is highlighted by red lines, while safe range (70-180 [mg/dl]) in brown.

Note that these inaccuracies in the meal management, as well as making the scenario more realistic, also increment the number of critical episodes due to a not optimal insulin therapy.

2.2.2. Data preprocessing

The glucose concentration measured by CGM sensors can be affected by sensor measurement noise and calibration errors typical of the devices used to measure the glucose subcutaneously. In order to minimize the effect of these problems, the data used to train the LSTM network are preprocessed. To have a more accurate signal, a preprocessing algorithm, called "retrofitting" [36], is used. This algorithm reconstructs the BG profile starting from CGM measurements and using sporadic BG values, if available, and Self-Monitoring Blood Glucose (SMBG) measurements, obtained by the patients from finger-stick. In the following, CGM will refer to the CGM after retrofitting.

Furthermore, because the involved quantities have different ranges, CGM, meals and insulin quantities are all rescaled using mean normalization. This is a typical approach to improve neural network training performance since unscaled data can cause a slow learning phase, non-convergence, or exploding gradient problems, particularly when working with RNNs.

Lastly, the output of the network is filtered through robust quadratic regression to eliminate the prediction's noise.

2.3. State of the art: Personalized LSTM for glucose prediction

In [28] a Personalized LSTM network, named P-LSTM, was trained for each of the 100 patients of the in silico adult population of the UVA/Padova simulator, using the *tr-dataset* and *v-dataset* described in Section 2.2. All the networks were trained using Adaptive Moment Estimation (Adam) optimizer to perform the backpropagation algorithm [31], with Mean Squared Error (MSE) as the loss function (L):

$$L = MSE = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2$$

where $\hat{y}$ is the predicted value, y is the real value and N is the number of samples. This minimization is an iterative process, where at each iteration (j), called epoch, the gradient of L with respect to the weights (w) is calculated, to update the weights in the opposite direction with respect to the gradient:

$$w(j+1) = w(j) - \alpha \frac{\partial}{\partial w} L\left(w(j)\right)$$

where α is the so-called learning rate, that determines the step size at each iteration. The P-LSTMs, composed by a single layer LSTM and a fully connected layer, were trained in [28] using the same network hyperparameters for all the patients: neurons $n_c = 96$, learning rate $\alpha = 0.01$ and number of epochs $n_e = 500$. The training was implemented with TensorFlow [37] and Keras API [38], the hyperparameters were optimized on a subset of the patients using KerasTuner [39], where the optimization criterion is the minimization of the L on the *v-dataset*. Moreover, a PH of 40 [min] was chosen in order to have a good

trade-off between clinical needs and network performances accordingly to the analysis conduced in [24] which found out that the best trade-off is obtained with a PH of 30 [min] but acceptable results are achieved with a PH of 45 [min]. The value was chosen also to be consistent with [28].

The P-LSTMs were evaluated through the following performance indexes on the *ts-dataset*, being the signals y, $\hat{y}$ and the parameter N the ones defined above, $\overline{y}$ the mean of y, and $\|\cdot\|$ the norm of the signal:

- Root Mean Square Error ($RMSE$) that quantifies the variance of the prediction error; the smaller it is, the better is the prediction. It is expressed as:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{2}$$

- Index of fitting (FIT), a normalized index that indicates how much the prediction matches the real data. For a perfect prediction it is equal to 100%, otherwise smaller and it can also be negative. It is expressed as:

$$FIT = 100\left(1 - \frac{\sum_{i=1}^{N}\|\hat{y}_i - y_i\|}{\sum_{i=1}^{N}\|y_i - \overline{y}_i\|}\right) \tag{3}$$

- Downward Delay (DD) and Upward Delay (UD), the time shifts that minimize the mean squared error between real and predicted data [40, 41] computed on negative and positive trends respectively, as follows:

$$\begin{aligned} DD &= \underset{j\in[0,PH]}{\arg\min}\left[\frac{1}{N}(\hat{y}(t|t-PH+j) - y(t))^2\right] \\ &\forall t \in [t_P, t_{N_{75}}] \\ UD &= \underset{j\in[0,PH]}{\arg\min}\left[\frac{1}{N}(\hat{y}(t|t-PH+j) - y(t))^2\right] \\ &\forall t \in [t_N, t_{P_{75}}] \end{aligned} \tag{4}$$

Since, as observed in [41], the delay computed on the entire signal results too pessimistic, Sparacino et al. propose to compute these indexes not using Peaks (P) and Nadirs (N) but new thresholds, P_{75} and N_{75} that are the 75% of P and N respectively. So the time instants at which

these thresholds are crossed ($t_N, t_P, t_{N_{75}}, t_{P_{75}}$) are identified and used in Equation (4). DD and UD are calculated on each single trend and then the average delays of the total thresholds crossings are computed. For clarity, an example of peaks, nadirs and the above-mentioned thresholds is shown in Figure 4.

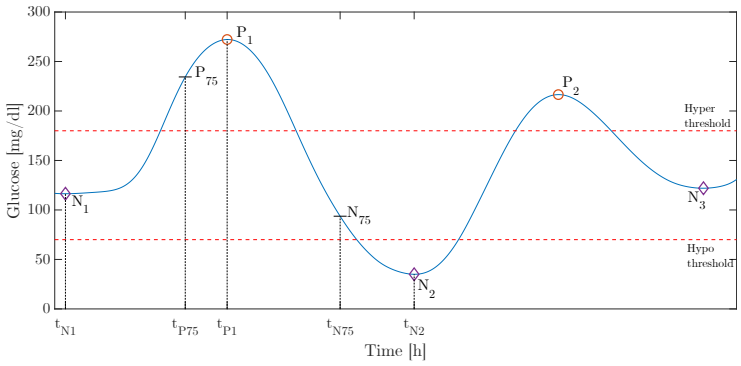


Figure 4: Example of glucose profile, where nadirs are pointed with purple diamonds, peaks with red circles and positive and negative threshold are highlighted. Red dashed lines show the hypoglycemia and hyperglycemia thresholds.

In [28] the P-LSTM showed promising results in terms of prediction capabilities with respect to the literature (RMSE below 8, FIT around 76% and delays around 9 minutes). In view of them, the authors used the P-LSTM to design a personalized hypoglycemia Alarm System [42]. The AS showed not satisfactory performances for some patients due to the poor prediction ability of some P-LSTM trained with fixed hyperparameters ($n_c = 96$, $\alpha = 0.01$ and $n_e = 500$). Then, in this work a new version of the P-LSTM model, called Enhanced Personalized LSTM model (EP-LSTM) is proposed. Since the goal of this work is to use LSTM models to design personalized alarm systems that predict both hypoglycemia and hyperglycemia events (and not only hypoglycemia as in [42]), the prediction performances have to be optimized. In particular, the delay with respect to the original signal has to be limited in order to catch the glucose dynamics and the peaks have to be well reproduced. In this prospective, the optimization of the hyperparameters used to train the P-LSTM is performed

via KerasTuner on the specific patient giving rise to the EP-LSTM.

In Figure 5 a schematic representation of the proposed work is reported: the data generation is performed through the UVA/Padova simulator exploiting a MPC controller to define the patient therapy; the EP-LSTM is then trained for each patient optimizing the hyperparameters. Finally, the AS is designed using the EP-LSTM and evaluated on a new dataset generated through the same simulator in different conditions to check the capability to detect in advance adverse conditions for the patient, allowing the prevention of upcoming hypo- and hyperglycemia phenomena. It is important to note that the data collection and acquisition can be substituted with in vivo test acquiring data from real patients. Since in vivo data are not available for this work, the authors selected the most realistic metabolic simulator for T1D patients (see Appendix A).

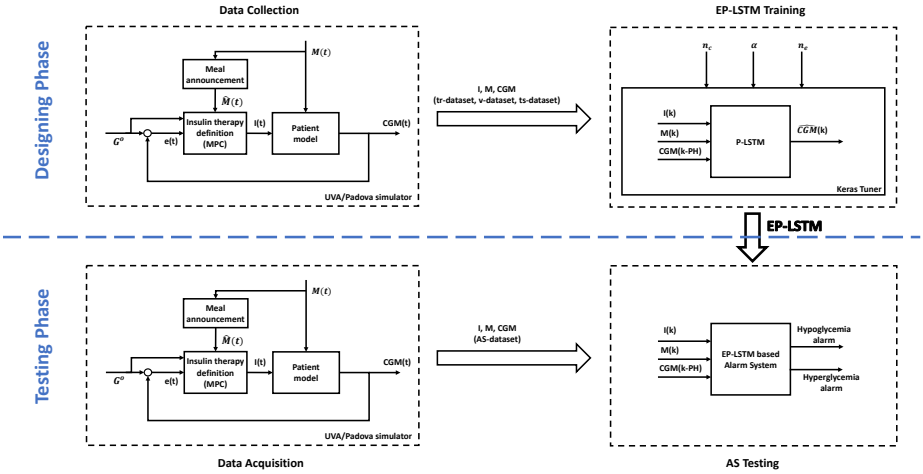


Figure 5: Diagram representing the proposed approach to design a personalized hypo/hyperglycemia AS.

3. EP-LSTM based Alarm System

3.1. Alarm system description

Alarm systems are powerful tools that can be used to avoid harmful situations like hypoglycemia and hyperglycemia for subjects with T1D. The ASs

can be of two different types, threshold detection or prediction-based alarms, depending on the way the alert is generated. Threshold detection alarms are triggered when certain critical thresholds are crossed, meaning in this particular application when critical BG levels are reached (70 [mg/dl] for hypoglycemia, 180 [mg/dl] for hyperglycemia). Instead, prediction-based alarms exploit a model to predict the BG levels and, if in the near future the prediction is below the hypoglycemia threshold or above the hyperglycemia one, the patient is alerted in advance to take actions in order to prevent the upcoming critical event. For this kind of alarms, the quality of the predictions plays a key role in the performance of the entire AS [43, 44].

In this work the EP-LSTM is proposed as core of a new personalized prediction-based AS. The AS considers the EP-LSTM predictions at time k^* over a Prediction Window (PW) from k^*+1 to k^*+PW: if the predicted BG levels exceeds the threshold G_{hypo} (or G_{hyper}) for at least S minutes, an hypoglycemia (or hyperglycemia) alarm is raised. In this way the patient is alerted of an incoming critical event and can take some action to avoid it. Note that $PW \leq PH$ since the EP-LSTMs require CGM in the previous PH minutes to predict future CGM; in this work we decided to exploit the longest PH which the EP-LSTM was trained for.

3.2. Data generation for AS evaluation

In order to test the AS performances, a new dataset is generated, called *AS-dataset*. *AS-dataset* contains three meals per day with the possibility to have snacks and uncertainties in the meal announcement are considered accordingly to the other datasets. There are not repetitions of the days that are also uncorrelated with the scenarios of *tr-dataset*, *v-dataset* and *ts-dataset*. *AS-dataset* is described in detail in Table 2. In order to effectively test the AS performances, a relevant amount of hypoglycemia and hyperglycemia events distributed over this time period is needed, so *AS-dataset* is much longer than the others to remain realistic. The MPC used in the UVA/Padova simulator has the same paramet-

Dataset	Meals	CHO (g)	Time
	Breakfast	55 [50-60]	7:30 [7:00-7:45]
AS-dataset	Lunch	65 [60-65]	13:00 [12:30-13:45]
(28 days of data)	Dinner	68 ± 12	19:30 [19:30-20:00]
	Snacks	20 [15-30]	

Table 2: *AS-dataset* description

ers used for the training dataset in order to represent the patients undergoing to the same insulin therapy. Additional details are reported in Appendix A.

3.3. Performance metrics for alarm systems

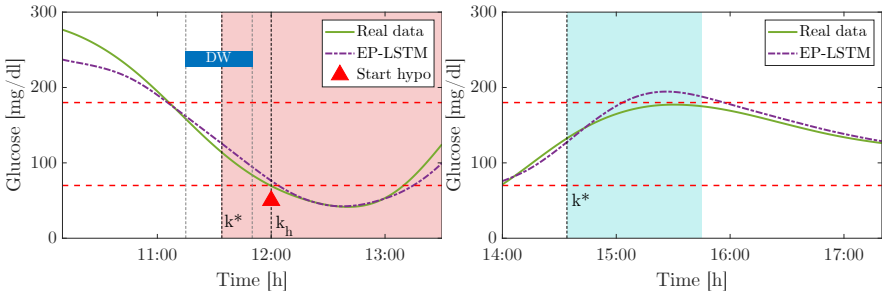
A preliminary definition of the events that can occur during hypo- or hyperglycemia prediction is needed to describe the performance metrics for the evaluation of the alarm goodness. Firstly, k_h is defined as the time instant at which a generic hypo- or hyperglycemia event occurs; DW is the Detection Window determined by the event, starting DW_s (Detection Window start) and ending DW_e (Detection Window end) minutes before the considered time, with $DW_s > DW_e > 0$ to guarantee the alarm diagnostic, where DW_e is the minimum interval to detect the event; k^* the time instant at which a generic hypo- or hyperglycemia alarm is raised. The following events are valid both for hypo- and hyperglycemia detection:

- True Positive (TP): at time k_h an event occurs and an alarm is raised in the Detection Window, $DW = [k_h - DW_s, k_h - DW_e]$, as shown in Figure 6a for hypoglycemia. In practice, the alarm must be activated at least DW_e minutes before the event to allow the patient to prevent it and also not too distant in time, at most DW_s, to be considered related to the current event.

- False Positive (FP): at time k^* an alarm is raised but there is no event in $[k^*, k^* + DW_s]$, as shown in Figure 6b for hyperglycemia. It is important

to note that if an alarm is raised too late, after DW_e, it is not considered a FP event even if it is not counted as a TP. In fact, it is a correct detection of the event (TP) that is not raised enough in advance to take an action to avoid hyper/hypoglycemia. In this prospect, those events are not taken into account in the results.

- False Negative (FN): at time k_h there is an event but the alarm is not activated in the DW, as in Figure 6c for hypoglycemia.
- True Negative (TN): at time k^* the alarm is not activated and there is no event in the interval $[k^*, k^* + DW_s]$, as in Figure 6d.



(a) Example of TP

(b) Example of FP

(c) Example of FN

(d) Example of TN

Figure 6: Example of TP, FP, FN and TN events. In each plot the real data are in green, prediction through EP-LSTM in dotted purple, red triangle points the start of an hypoglycemia event and its detection window. The alarm activation is represented by the light-red region in the hypoglycemia case and by light-blue region in the hyperglycemia case. Red dashed lines show the hypo- and hyperglycemia threshold.

Considering these events, the following metrics to evaluate the AS performances are defined:

- True Positive Rate (TPR), also called recall or sensitivity, measures how many positive events are properly detected over the total amount of events:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

- Positive Predicted Value (PPV), or precision, measures how many alarms are correctly activated over the total amount of alarms rising:

$$\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

- F1 score (F1), the harmonic mean of TPR and PPV:

$$\mathrm{F1} = \frac{2 \cdot \mathrm{TPR} \cdot \mathrm{PPV}}{\mathrm{TPR} + \mathrm{PPV}}$$

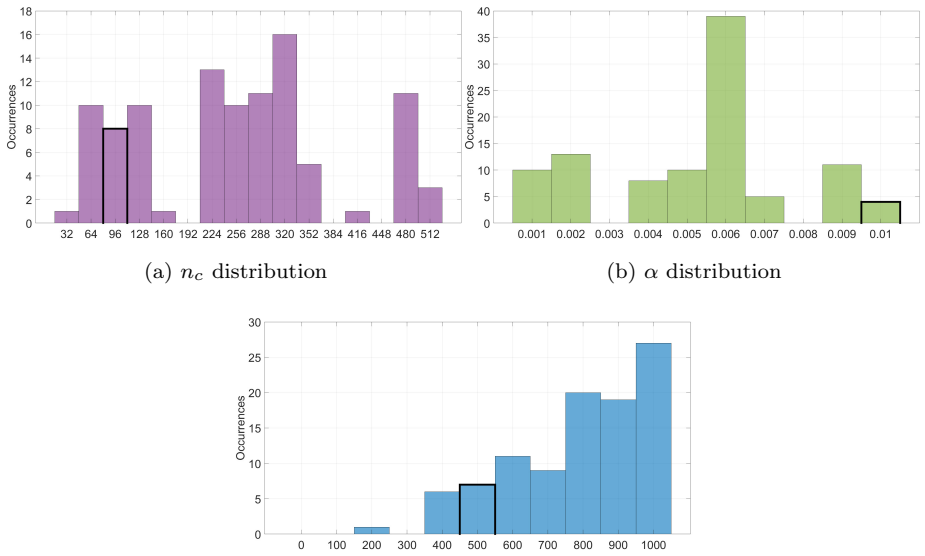
It should be noted that only the metrics related to the positive events are here considered because the dataset is very unbalanced, having few hypoglycemia and hyperglycemia events with respect to the overall amount of data in the safe range. In fact, the number of TN events is definitely higher than the others and all the metrics including it would be saturated and misleading. This is proven in [45], showing that for unbalanced sets precision and recall are more informative than ROC plots, where TPR is compared with its counterpart, False Positive Rate (FPR=FP/(FP+TN)).

4. Results

4.1. EP-LSTM results

We use a computer equipped with graphics processing unit (GPU) to carry out the training processes: an Intel i9-10920X CPU with 3.50 GHz, 32.0 GB memory with GPU NVIDIA GeForce RTX 2080 Ti was used in the experiments. The program was written in Python 3.6, using CUDA 9.0.

A specific hyperparameters tuning is performed for the 100 adult patients of the UVA/Padova simulator via KerasTuner.



(a) n_c distribution

(b) α distribution

(c) n_e distribution

Figure 7: Hyperparameters distributions in the 100 adult patients. Original hyperparameters values used for the P-LSTM training are highlighted with black thick edges.

The total distributions of the hyperparameters are presented in Figure 7, where three bar plots show the occurrences of the values for each hyperparameter: n_c in Figure 7a, α in Figure 7b and n_e in Figure 7c. It can be noticed that the original values resulted optimal only for a limited number of patient, justifying the optimization procedure used for the EP-LSTM.

After the retuning of the hyperparameters, significant improvements can be noticed evaluating the indexes: on the *v-dataset* DD improved from 9.75 [min] to 9.18 [min] ($p < 0.05$) and UD from 10.17 [min] to 8.81 [min] ($p < 0.001$), RMSE was reduced from 6.49 [min] to 4.68 [min] ($p < 0.001$) and FIT was improved from 76.76 [min] to 82.35 [min] ($p < 0.001$). The p-values are computed with the appropriate statistical test based on the data distribution characteristics. The gaussianity and homoscedasticity of the data distributions are assessed by

the Lilliefors test and two-sample F-test, respectively. If at least one distribution is non-Gaussian, the Wilcoxon rank sum test is used to test the significance of the differences; if both distributions are Gaussian and homoscedastic, a two-sample t-test is performed; otherwise, the two-sample t-test with Satterthwaites approximation is used.

The 100 EP-LSTMs are then evaluated through RMSE, FIT, DD and UD on the *ts-dataset* and the results are reported in Table 3 compared with the ones obtained with the P-LSTM. Mean (± SD) or median [25^{th} - 75^{th}] percentiles are reported for the entire population, if the results are normally or not normally distributed, respectively. The *ts-dataset* DD was improved from 9.87 [min] to 9.71 [min] ($p > 0.05$) and UD from 9.28 [min] to 8.31 [min] ($p < 0.001$), RMSE was reduced from 7.80 [min] to 6.45 [min] ($p < 0.001$) and FIT was improved from 75.67 [min] to 79.40 [min] ($p < 0.001$).

Index	EP-LSTM	P-LSTM
RMSE	6.45^a [5.28, 7.43]	7.80 (±1.98)
FIT	79.40^a [75.20, 82.80]	75.67 [70.72, 79.01]
DD	9.71 (±2.52)	9.87 (±3.17)
UD	8.31^a (±3.11)	9.28 (±3.71)

Table 3: Results of the prediction on the test dataset for EP-LSTM and P-LSTM. (^{a}p-value < 0.001, ^{b}p-value < 0.01, ^{c}p-value < 0.05)

The EP-LSTMs obtain overall a significant improvement with respect to the previous results, guaranteeing more reliability of the models for their employment in the alarm system. A graphical example is reported in Figure 8 for a case study: P-LSTM and EP-LSTM models are compared on the *ts-dataset*. It can be noticed that the EP-LSTM prediction is more accurate, especially in the hypoglycemia and hyperglycemia regions, and the delays are reduced: for this patient, DD goes from 18.57 [min] to 6.27 [min] and UD from 22.60 [min] to

14.73 [min].

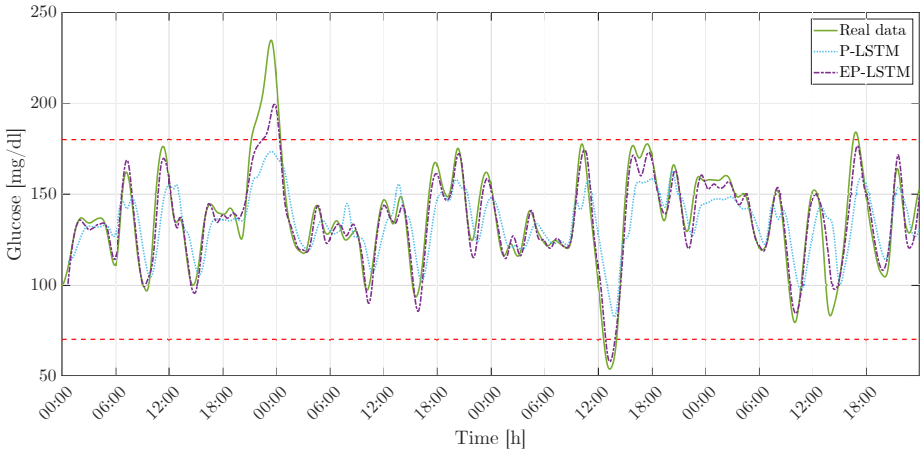
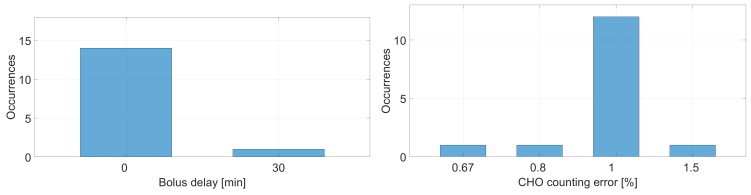


Figure 8: Glucose profile of a subject, in green the real data, in blue dotted the P-LSTM prediction, in purple dashed the EP-LSTM prediction. Red dashed lines are hyperglycemia and hypoglycemia thresholds.

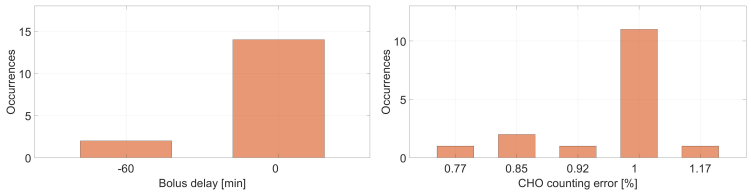
4.2. AS results

The AS performances are evaluated on the *AS-dataset* for all the 100 adult patients of the UVA/Padova simulator. In this work, the AS was designed setting $DW_s = 45$ [min] and $DW_e = 10$ [min], in accordance with previous literature [43, 44], and PW=40 [min], $G_{hypo} = 70$ [mg/dl], $G_{hyper} = 180$ [mg/dl], S=10 [min]. Among the 100 adult patients, 3 subjects did not present hyperglycemia events in the *AS-dataset* so for the hyperglycemia alarm the results are calculated on 97 patients. The median $[25^{th} - 75^{th}]$ percentiles is then calculated on the entire population since the results are not normally distributed. The performances of the AS are shown in Table 4: the EP-LSTM AS obtained satisfactory performances with the 77% of hypoglycemia events correctly detected (TPR=77%) over an average number of episodes equal to 12 and with the 89% of hyperglycemia events correctly detected (TPR=89%) over an average number of episodes equal to 39. The precision of the alarm is also adequate with an 83% of the hypoglycemia alarms raised correctly (PPV=83%) and an 85% of the hyperglycemia alarms raised correctly (PPV=85%). The improve-

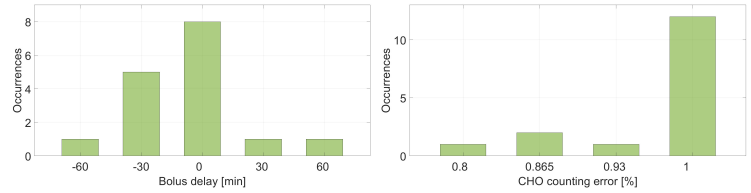
ments obtained with the hyperglycemia AS are statistically significant with a mean of 7% as reported in Table 4. A graphical example of the AS performances is shown in Figure 9, where the first four days of the *AS-dataset* on a specific patient are shown.



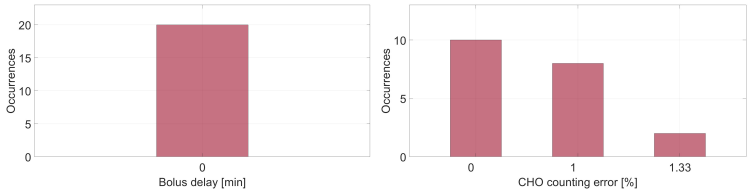
(a) Breakfast



(b) Lunch



(c) Dinner



(d) Snack

Figure 3: Histograms representing the time delay and the CHO counting error of the insulin boluses delivered to compensate different type of meals.

Index	EP-LSTM		P-LSTM		Improvement	
	Hypo Alarm	Hyper Alarm	Hypo Alarm	Hyper Alarm	Hypo Alarm	Hyper Alarm
TPR	76.92 [64.29, 95.65]	89.13^{b} [73.86, 97.69]	75.00 [57.33, 88.89]	82.61 [70.05, 96.26]	2.56	7.89
PPV	83.33 [71.43, 100.00]	85.29^{a} [73.09, 90.93]	78.38 [66.67, 92.86]	79.03 [62.35, 88.26]	6.32	7.92
F1	78.79 [66.67, 87.05]	83.87^{a} [71.91, 91.89]	75.00 [66.67, 84.46]	80.6452 [62.26, 89.26]	5.05	4.00
# episodes	12 $\pm$ 9	39 $\pm$ 26	12 $\pm$ 9	39 $\pm$ 26		
# alarms	12 $\pm$ 9	40 $\pm$ 27	12 $\pm$ 9	40 $\pm$ 27		

Table 4: Hypoglycemia and hyperglycemia alarm systems performances with EP-LSTM and P-LSTM. (^{a}p-value < 0.001, ^{b}p-value < 0.01, ^{c}p-value < 0.05)

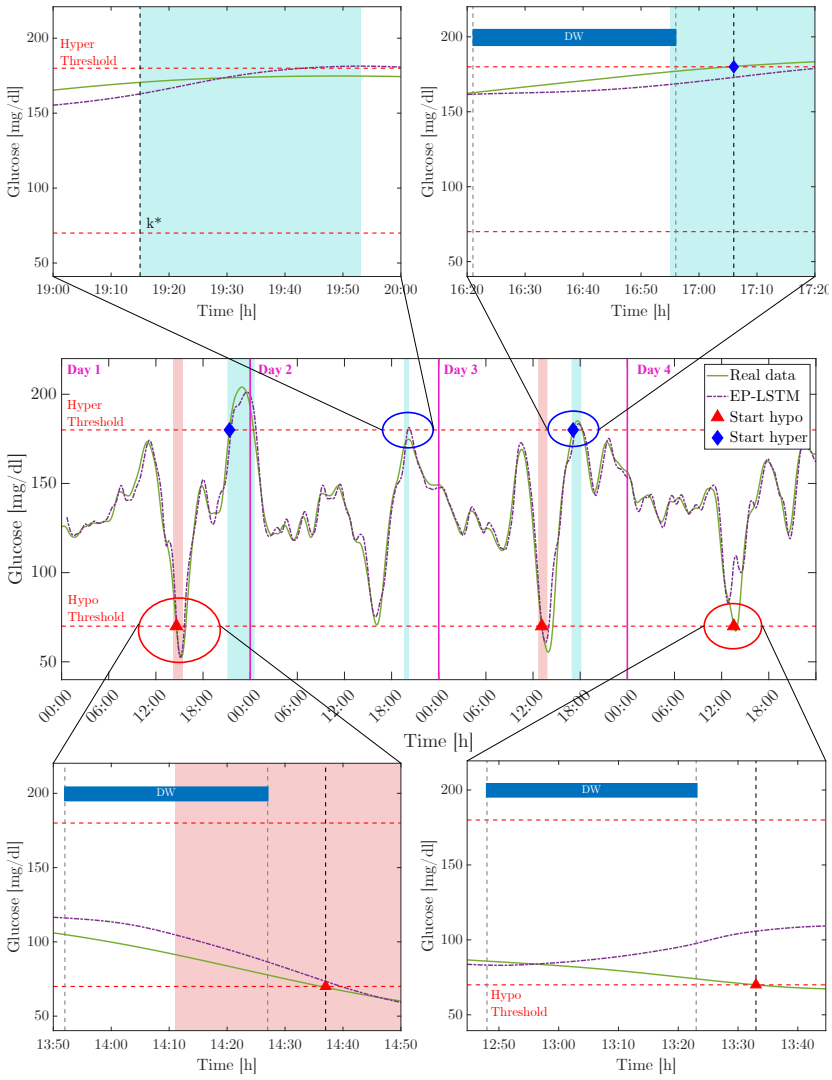


Figure 9: Personalized alarm system example. The real data are in green, prediction through EP-LSTM in dotted purple, red triangle (blue diamonds) points the start of an hypoglycemia (hyperglycemia) event and its detection window. The alarm activation is represented by the light-red (light-blue) region in the hypoglycemia (hyperglycemia) case.
Four cases are highlighted in the zoomed panels: FP (left) and TP (right) for hyperglycemia event on the top, while TP (left) and FN (right) for hypoglycemia on the bottom.

In the middle panel the real data and the predictions are represented, among with hypoglycemia and hyperglycemia events. The alarms activation is repres-

ented by the light-red (light-blue) regions for hypoglycemia (hyperglycemia). During the four days, three hypoglycemia and two hyperglycemia episodes are present: four of them are correctly detected, while around 14:00 of Day 4 a FN event for hypoglycemia can be noticed and around 19:00 of Day 2 a FP event for hyperglycemia is present. In the bottom panel two zooms in of two hypoglycemia episodes are shown, highlighting how the alarm is correctly raised in the DW on the left (TP) while the event is not detected on the right (FN). In the top panel a wrong hyperglycemia detection is presented on the left (FP) while a corrected one is represented on the right (TP). From two of the zoomed panels, it can be observed that these false events are not severe or harmful for the patient. In fact, the hypoglycemia event that is not detected is not a severe event (BG $>$ 60 [mg/dl]), while the hyperglycemia alarm is wrongly raised when the glucose was closed to the hyper threshold (BG $>$ 170 [mg/dl]) as shown in Figure 6. In general the results obtained from the proposed AS can be considered satisfying. A possible improvement could be related to the hypoglycemia AS increasing the number of episodes used for the training of the EP-LSTM in order to significantly improve also its performance.

In order to have an idea of the typical performance indexes obtained in literature by this kind of ASs, a comparison can be made with a recent study [40], where thirty different linear and non-linear algorithms for BG prediction are used to create an hypoglycemia prediction-based alarms. The same real data are used to create both population and personalized models and the results of the alarm system are evaluated in terms of TPR and PPV. The authors found that the best linear technique was the individualized ARIMA with TPR of 82% and PPV of 64%, while the best non-linear one was the support vector regression with TPR of 69% and PPV of 63%. An LSTM population model was considered too and the results were worse, with TPR of 26% and PPV of 46%. Above all, the presented EP-LSTM AS presents promising performances with respect to the ones obtained in literature on in vivo data; however a fair comparison is not possible since this work is based on in silico data, while [40] uses in vivo data, that is much more challenging.

5. Conclusions

In this work a personalized alarm system for hypoglycemia and hyperglycemia prevention based on personalized neural network models is presented. The LSTM networks are trained for each patient of the in silico population of the UVA/Padova simulator, in order to predict future CGM values considering meals, insulin and past CGM values in input, with a prediction horizon of 40 minutes. Starting from the results presented in literature, where a personalized LSTM with population hyperparameters was proposed, in this work the specific patient hyperparameters are ad-hoc optimized. The obtained results show RMSE of 6.45 and FIT of 79.40%, with delays of around 9 minutes. Then, these models are used by an alarm system for both hypo- and hyperglycemia prediction. The performances of the alarm system are satisfying, with TPR of 76.92%, PPV of 83.33% and F1 of 78.79% for hypoglycemia detection and TPR of 89.13%, PPV of 85.29% and F1 of 83.87% for hyperglycemia detection. Seen these promising results, these particular alarm systems can be successfully included in SAP systems or APs to increase their efficiency and further reduce patients distress.

The main limitation of this work is related to the use of in silico data. A future development could be the inclusion of the inter-day variability in the simulator in addition to the intra-day variability already considered, in order to test the proposed system on a more challenging and realistic case. It is worth to be noted that the presented procedure can be also applied to real patients since it does not require invasive measurements nor clinical trials. A deeper analysis of the alarm system in the hyperglycemia case can be conducted, trying different PHs in order to anticipate the DW of the alarm, since in this case the current DW could be not enough to avoid the upcoming episode. Moreover, a cluster analysis of the patients could be included in order to define groups of patients that share the same clinical parameters and could share also the same control and, the network hyperparameters. In the end, this methodology can be extended on real data collected during clinical trials. This can be a challenging

application considering all the uncertainties present in real-life data but it results very useful to help T1D patients dealing with the harmful situations that can occur due to their pathology.

Appendix A. Simulation settings

UVA/Padova simulator

The availability of glucose-insulin models that simulate the glucose response to insulin and meal intake is a key point for the design and evaluation of glucose sensors, control algorithms, alarms and decision support systems. In fact, computer simulations allow to perform several in silico tests with relevant time- and cost- savings, and they also allow to simulate experiments potentially dangerous for patient safety.

In the last decades, several simulation tools have been developed (see [46] for a review), each one based on a comprehensive mathematical model and equipped with an in silico population. In 2008 the US Food and Drug Administration (FDA) accepted the UVA/Padova simulator, a T1D simulator developed by Universities of Virginia (UVA) and Padova, as a substitute for preclinical trials for closed-loop algorithms tests. This simulator is equipped with 300 in silico subjects generated from a joint distribution of model parameters, obtained by identifying a complex model [47] from a unique multiple-tracer dataset. Each patient is represented by a set of parameters extracted from this distribution. This simulator is able to span the variability observed in the real T1D population representing the inter-individual variability that characterizes this population. Moreover, the new version of the simulator [6] incorporates a nonlinear model to describe the glucose response to hypoglycemia and the counter-regulation [48], and a model with time-varying parameters to describe the intra-day variability of the Insulin Sensitivity (SI) and the meal intestinal absorption rate, taking into account the circadian variability of SI and the dawn phenomena [49]. The complete model equations can be found in [6]. In summary, the final version of the simulator used in this work includes both the inter- and intra-subject

variability typical of the population under study making the results reliable. This simulator has been used by more than 30 sites in academia and companies involved in T1D research, more than 70 articles were published in peer-reviewed journals [6]. The Epsilon Group (TEG) offers a commercial version of this simulator, the TEG's Diabetes Mellitus Metabolic Simulator for Research (DMMS.R). It provides a unique in silico environment for testing diabetes treatment and monitoring interventions, and it is ideal for modeling new devices or examining treatment protocols and dosing algorithms. For this reason, the data generated by the simulator and the parameters that define the 100 subjects can not be made publicity available.

CGM noise model

The measurement error model used in this work to simulate the CGM measurements on the UVA/Padova simulator is the autoregressive model proposed in [34], which was identified exploiting 141 datasets relative to the 47 patients enrolled in the CAT AP@home trial [50]. This model describes the total measurement error, including wearing issues in addition to noise and drift usually considered. It can be described by the formula:

$$\varepsilon_{PV}(k) = a_1 \cdot \varepsilon_{PV}(k-1) + a_2 \cdot \varepsilon_{PV}(k-2) + v(k)$$

where the error $v(k)$ is a Gaussian White Noise with mean μ_v and variance σ_v^2; the parameters a_1, a_2 are coefficients of the AR model and the distribution of the initial states of the process is normal with mean μ_{is} and variance σ_{is}^2. The parameters used in this work are: $a_1 = 1.5458$, $a_2 = -0.5708$, $\mu_v = 0.0017$ [mmol/L], $\sigma_v^2 = 0.0283$ [mmol^2/L^2], $\mu_{is} = [-0.1766 \quad -0.1566]$ [mmol/L] and

$$\sigma_{is}^2 = \begin{bmatrix} 0.7759 & 0.7895 \\ 0.7895 & 0.8603 \end{bmatrix} \quad [\text{mmol}^2/\text{L}^2].$$

Pavia MPC controller

The insulin therapy in all datasets is computed by the MPC developed in [34] tuned in a sub-optimal way in order to have a significant amount of critical

episodes. In fact, the goal of this work is the detection of hypoglycemia and hyperglycemia events, so we need to simulate patients with regulation problems. The MPC algorithm [34] is a Linear-Model-Predictive-Control (LMPC) that uses an approximate linear model of the insulin-glucose dynamics in order to predict the future glucose profile given the carbohydrates and insulin intakes. This model is obtained by the linearization around a suitable working point of the more complex nonlinear Dalla Man model [51] of the average in silico patients of the UVA/Padova simulator. This linearized model can be written as:

$$\begin{cases} x\left(k+1\right) = Ax\left(k\right) + Bu\left(k\right) + Md\left(k\right) \\ y\left(k\right) = Cx\left(k\right) \end{cases} \tag{A.1}$$

where

- $x(k) \in R^{13}$, is the 13-state vector;
- $y(k) = CGM(k) - G_b$ [mg/dl] is the difference between the subcutaneous glucose CGM and the basal value (G_b);
- $u(k) = i(k) - u_b(k)$ [pmol/kg] is the difference between the injected insulin i and its reference basal value u_b, normalized by the patient weight;
- $d(k)$ [mg], represents the meal.

The triplet (A, B, C) is assumed both stabilizable and detectable. The MPC cost function is a quadratic penalty function J defined as follows:

$$J\left(x\left(k\right), u\left(\cdot\right), k\right) = \sum_{i=0}^{N-1} \left(q\left(y\left(k+i\right) - y_o\left(k+i\right)\right)^2 + \left(u\left(k+i\right) - u_o\left(k+i\right)\right)^2\right) + \|x(k+N)\|_P^2 \tag{A.2}$$

where q is the positive scalar weight and N is the prediction horizon. Moreover, $||\ x(k+N)\ ||_P = x(k+N)'Px(k+N)$, where P is the unique nonnegative solution of the discrete time Riccati equation

$$P = A'PA + qC'C - A'PB(1 + B'PB)B'PA$$

The reference signals are defined as:

- $y_o(k) = \widetilde{y}(k) - G_b$ [mg/dl], is the difference between the reference value ($\widetilde{y}$) of the subcutaneous glucose and the glucose basal value;

- $u_o(t) = \widetilde{u}(k) - u_b(k)$ [pmol/kg], is the difference between the reference value ($\widetilde{u}$) of the insulin profile and the insulin basal value, normalized by the patient weight.

The proposed algorithm does not include explicit constraints, so it is possible to calculate the closed form solution as follows

$$u^{MPC}(k) = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} (-K_x x(k) - K_d D(k) + K_{Y_o} Y_o(k) + K_{U_o} U_o(k))$$

where $K_x \in R^{Nx13}$, $K_d \in R^{NxN}$, $K_{Y_o} \in R^{NxN}$, $K_{U_o} \in R^{NxN}$ are derived as described in [52] and

$$D(k) = \begin{bmatrix} d(k) & \dots & d(k+N-2) & d(k+N-1) \end{bmatrix}'$$

is the vector of future meals and

$$Y_o(k) = [y_o(k+1) \quad \cdots \quad y_o(k+N-1) \quad 0]'$$

$$U_o(k) = [u_o(k) \quad \cdots \quad u_o(k+N-2) \quad u_o(k+N-1)]'$$

Since the state $x(k)$ of the model is not accessible, a Kalman Filter is used to estimate it. The linear system A.1 can be written considering the noises as:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) + Md(k) + v_x(k) \\ y(k) = Cx(k) + v_y(k) \end{cases}$$

where $v = [v_x \;\; v_y]$ is a multivariate zero-mean white Gaussian noise with covariance matrix:

$$V = \begin{bmatrix} Q_{KF} & 0 \\ 0 & R_{KF} \end{bmatrix}, \; Q_{KF} > 0, \; R_{KF} > 0$$

and the initial state $x_0 = x(0)$ is assumed to be a zero mean Gaussian random variable independent of v.

The control weight q in Equation A.2 is individualized using body weight, BW, and CR ratio following the formula:

$$q = Q_m \; e^{(-0.0366 * BW \; -0.2149 * CR + \; 2.5444)}$$

This algorithm with $Q_m = 1$ was successfully used in 3 outpatient trials lasted 1-2 months [53–55], while in this work the parameter Q_m is set equal to 10 to make sub-optimal the control action, in order to have a significant amount of critical episodes (hypo/hyperglycemia events) in all the datasets.

Scenarios

The details of the three scenarios used in this work are reported in Tables A.5-A.7. The number of meals per day and the time distribution are inspired by real data distribution [35] and to provide more realistic settings during the data generation phase, the meal announcement presents inaccuracies in terms of amount (CHO) and time of the carbohydrate intake. For each datatset, time and quantities of meals are reported in the tables for each day in the first columns, then the last one reports if the insulin bolus is delivered at the same time of the meal and if the meal quantity used for its computation is correct or not.

	Time	CHO (g)	Insulin Bolus
Day 1	09:00	40	Bolus on time
	12:30	70	Bolus on time for 60 g
	19:30	60	Bolus at 19:00
Day 2	08:00	50	Bolus at 7:30
	13:00	80	Bolus on time
	17:00	30	Bolus on time for 40 g
	20:30	60	Bolus on time
Day 3	07:00	45	Bolus at 7:30
	12:00	60	Bolus on time for 70 g
	18:00	25	No bolus
	22:00	55	Bolus on time
Day 4	10:00	20	Bolus on time
	12:00	65	Bolus on time for 60 g

	19:00	90	Bolus at 19:30
Day 5	07:00	40	Bolus on time for 60 g
	12:30	65	Bolus on time
	20:00	70	Bolus at 19:30
Day 6	01:00	25	Bolus on time
	08:00	60	Bolus on time for 40 g
	13:00	60	Bolus on time
	20:30	55	Bolus at 21:00
Day 7	01:00	15	Bolus on time for 20 g
	06:00	100	Bolus on time for 80 g
	15:00	60	Bolus on time
	21:00	30	Bolus on time
Day 8	05:00	75	Bolus on time
	12:00	40	Bolus on time
	17:00	35	No bolus
	21:00	60	Bolus at 20:00

Table A.5: Training Scenario

	Time	CHO (g)	Insulin Bolus
Day 1	07:30	55	Bolus on time
	11:00	25	No bolus
	13:00	65	Bolus on time
	19:30	75	Bolus on time for 70 g
	22:30	15	Bolus on time
Day 2	08:00	50	Bolus on time
	13:00	80	Bolus on time
	16:00	30	No bolus
	19:30	70	Bolus at 19:00 for 60 g
	22:00	15	No bolus
Day 3	07:00	50	Bolus on time
	09:30	25	No bolus
	13:30	60	Bolus on time
	16:00	25	Bolus on time
	21:00	80	Bolus at 20:30 for 70 g
	23:30	15	No bolus
Day 4	07:30	50	Bolus on time
	14:00	60	Bolus on time
	18:00	30	Bolus on time
	21:00	65	Bolus on time

Table A.6: Validation Scenario

	Time	CHO (g)	Insulin Bolus
Day 1	06:00	60	Bolus on time
	09:30	25	No bolus
	12:30	65	Bolus on time
	15:30	15	Bolus on time
	20:00	100	Bolus at 19:30 for 80 g
Day 2	07:30	50	Bolus on time
	13:00	70	Bolus on time
	16:00	20	No bolus
	19:30	70	Bolus on time
Day 3	09:00	65	Bolus on time
	11:00	20	Bolus on time
	14:00	65	Bolus at 13:00 for 50 g
	19:30	90	Bolus on time
	23:30	20	No bolus
Day 4	07:00	55	Bolus on time
	10:00	20	Bolus on time
	13:30	60	Bolus at 12:30 for 50 g
	20:30	85	Bolus on time

Table A.7: Testing Scenario

CRediT authorship contribution statement

Francesca Iacono: Methodology, Validation, Software, Formal analysis, Data curation, Writing - Original draft, Writing - Review&editing **Lalo Magni:** Conceptualization, Writing - Review&editing. **Chiara Toffanin:** Methodology, Formal analysis, Data curation, Writing - Original draft, Writing - Review&editing, Supervision, Project administration.

Declaration of competing interest

The authors state that they have no known competing interests.

Acknowledgements

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] International diabetes federation, IDF Diabetes Atlas, 10th edn. Brussels, Belgium: International Diabetes Federation (2021).

[2] A. Janež, C. Guja, A. Mitrakou, N. Lalic, T. Tankova, L. Czupryniak, A. G. Tabák, M. Prazny, E. Martinka, L. Smircic-Duvnjak, Insulin therapy in adults with type 1 diabetes mellitus: a narrative review, Diabetes Therapy 11 (2) (2020) 387–409.

[3] R. M. Bergenstal, W. V. Tamborlane, A. Ahmann, J. B. Buse, G. Dailey, S. N. Davis, C. Joyce, T. Peoples, B. A. Perkins, J. B. Welsh, et al., Effectiveness of sensor-augmented insulin-pump therapy in type 1 diabetes, New England Journal of Medicine 363 (4) (2010) 311–320.

[4] J. Hermanides, K. Nørgaard, D. Bruttomesso, C. Mathieu, A. Frid, C. M. Dayan, P. Diem, C. Fermon, I. Wentholt, J. Hoekstra, et al., Sensor-augmented pump therapy lowers HbA1c in suboptimally controlled Type 1 diabetes; a randomized controlled trial, Diabetic Medicine 28 (10) (2011) 1158–1167.

[5] K. Nørgaard, A. Scaramuzza, N. Bratina, N. M. Lalić, P. Jarosz-Chobot, G. Kocsis, E. Jasinskiene, C. De Block, O. Carrette, J. Castañeda, et al., Routine sensor-augmented pump therapy in type 1 diabetes: the INTERPRET study, Diabetes technology & therapeutics 15 (4) (2013) 273–280.

[6] R. Visentin, E. Campos-Náñez, M. Schiavon, D. Lv, M. Vettoretti, M. Breton, B. P. Kovatchev, C. Dalla Man, C. Cobelli, The UVA/Padova type 1 diabetes simulator goes from single meal to single day, Journal of diabetes science and technology 12 (2) (2018) 273–281.

[7] S. Del Favero, C. Toffanin, L. Magni, C. Cobelli, Deployment of modular MPC for type 1 diabetes control: the Italian experience 2008–2016, in: The Artificial Pancreas, Elsevier, 2019, pp. 153–182.

[8] P. Abuin, P. S. Rivadeneira, A. Ferramosca, A. H. González, Artificial pancreas under stable pulsatile MPC: Improving the closed-loop performance, Journal of Process Control 92 (2020) 246–260.

[9] B. Kovatchev, Automated closed-loop control of diabetes: the artificial pancreas, Bioelectronic Medicine 4 (1) (2018) 1–12.

[10] R. Hovorka, V. Canonico, L. J. Chassin, U. Haueter, M. Massi-Benedetti, M. O. Federici, T. R. Pieber, H. C. Schaller, L. Schaupp, T. Vering, et al., Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes, Physiological measurement 25 (4) (2004) 905.

[11] D. Boiroux, V. Bátora, M. Hagdrup, S. L. Wendt, N. K. Poulsen, H. Madsen, J. B. Jørgensen, Adaptive model predictive control for a dual-hormone artificial pancreas, Journal of Process Control 68 (2018) 105–117.

[12] I. Hajizadeh, M. Rashid, A. Cinar, Plasma-insulin-cognizant adaptive model predictive control for artificial pancreas systems, Journal of process control 77 (2019) 97–113.

[13] D. Shi, E. Dassau, F. J. Doyle, Adaptive zone model predictive control of artificial pancreas based on glucose-and velocity-dependent control penalties, IEEE Transactions on Biomedical Engineering 66 (4) (2018) 1045–1054.

[14] E. Zijlstra, T. Heise, L. Nosek, L. Heinemann, S. Heckermann, Continuous glucose monitoring: quality of hypoglycaemia detection, Diabetes, Obesity and Metabolism 15 (2) (2013) 130–135.

[15] A. Weisman, J.-W. Bai, M. Cardinez, C. K. Kramer, B. A. Perkins, Effect of artificial pancreas systems on glycaemic control in patients with type 1 diabetes: a systematic review and meta-analysis of outpatient randomised controlled trials, The lancet Diabetes & endocrinology 5 (7) (2017) 501–512.

[16] C. Pérez-Gandía, A. Facchinetti, G. Sparacino, C. Cobelli, E. Gómez, M. Rigla, A. de Leiva, M. Hernando, Artificial neural network algorithm for online glucose prediction from continuous glucose monitoring, Diabetes technology & therapeutics 12 (1) (2010) 81–88.

[17] J. Nordhaug Myhre, M. Tejedor, I. Kalervo Launonen, A. El Fathi, F. Godtliebsen, In-Silico Evaluation of Glucose Regulation Using Policy Gradient Reinforcement Learning for Patients with Type 1 Diabetes Mellitus, Applied Sciences 10 (18) (2020) 6350.

[18] T. Zhu, K. Li, P. Herrero, P. Georgiou, Deep learning for diabetes: a systematic review, IEEE Journal of Biomedical and Health Informatics 25 (7) (2020) 2744–2757.

[19] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (8) (1997) 1735–1780.

[20] F. A. Gers, J. Schmidhuber, F. Cummins, Learning to forget: Continual prediction with LSTM, Neural computation 12 (10) (2000) 2451–2471.

[21] Q. Sun, M. V. Jankovic, L. Bally, S. G. Mougiakakou, Predicting blood glucose with an LSTM and Bi-LSTM based deep neural network, in: 2018 14th Symposium on Neural Networks and Applications (NEUREL), IEEE, 2018, pp. 1–5.

[22] A. Aliberti, I. Pupillo, S. Terna, E. Macii, S. Di Cataldo, E. Patti, A. Acquaviva, A multi-patient data-driven approach to blood glucose prediction, IEEE Access 7 (2019) 69311–69325.

[23] E. M. Aiello, G. Lisanti, L. Magni, M. Musci, C. Toffanin, Therapy-driven deep glucose forecasting, Engineering Applications of Artificial Intelligence 87 (2020) 103255.

[24] J. Carrillo-Moreno, C. Pérez-Gandía, R. Sendra-Arranz, G. García-Sáez, M. E. Hernando, Á. Gutiérrez, Long short-term memory neural network for glucose prediction, Neural Computing and Applications 33 (9) (2021) 4191–4203.

[25] S. Mirshekarian, H. Shen, R. Bunescu, C. Marling, LSTMs and neural attention models for blood glucose prediction: Comparative experiments on real and synthetic data, in: 2019 41st annual international conference of the IEEE engineering in medicine and biology society (EMBC), IEEE, 2019, pp. 706–712.

[26] M. F. Rabby, Y. Tu, M. I. Hossen, I. Lee, A. S. Maida, X. Hei, Stacked LSTM based deep recurrent neural network with kalman smoothing for blood glucose prediction, BMC Medical Informatics and Decision Making 21 (1) (2021) 1–15.

[27] C. Zecchin, A. Facchinetti, G. Sparacino, C. Cobelli, How much is short-term glucose prediction in type 1 diabetes improved by adding insulin delivery and meal content information to CGM data? A proof-of-concept study, Journal of diabetes science and technology 10 (5) (2016) 1149–1160.

[28] F. Iacono, L. Magni, C. Toffanin, Personalized LSTM models for glucose prediction in Type 1 diabetes subjects, in: 2022 30th Mediterranean Conference on Control and Automation (MED), 2022, pp. 324–329.

[29] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning representations by back-propagating errors, nature 323 (6088) (1986) 533–536.

[30] S. Hochreiter, The vanishing gradient problem during learning recurrent neural nets and problem solutions, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 6 (02) (1998) 107–116.

[31] P. J. Werbos, Backpropagation through time: what it does and how to do it, Proceedings of the IEEE 78 (10) (1990) 1550–1560.

[32] R. Pascanu, T. Mikolov, Y. Bengio, On the difficulty of training recurrent neural networks, in: International conference on machine learning, PMLR, 2013, pp. 1310–1318.

[33] R. Visentin, C. Dalla Man, B. Kovatchev, C. Cobelli, The university of Virginia/Padova type 1 diabetes simulator matches the glucose traces of a clinical trial, Diabetes technology & therapeutics 16 (7) (2014) 428–434.

[34] C. Toffanin, M. Messori, F. Di Palma, G. De Nicolao, C. Cobelli, L. Magni, Artificial pancreas: model predictive control design from clinical experience, Journal of diabetes science and technology 7 (6) (2013) 1470–1483.

[35] E. M. Aiello, C. Toffanin, L. Magni, G. De Nicolao, Model-based identification of eating behavioral patterns in populations with type 1 diabetes, Control Engineering Practice 123 (2022) 105128.

[36] S. Del Favero, A. Facchinetti, G. Sparacino, C. Cobelli, Improving accuracy and precision of glucose sensor profiles: retrospective fitting by constrained deconvolution, IEEE Transactions on Biomedical Engineering 61 (4) (2013) 1044–1053.

[37] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, software available from tensorflow.org (2015).
URL `https://www.tensorflow.org/`

[38] F. Chollet, et al., Keras, `https://keras.io` (2015).

[39] T. O'Malley, E. Bursztein, J. Long, F. Chollet, H. Jin, L. Invernizzi, et al., Kerastuner, `https://github.com/keras-team/keras-tuner` (2019).

[40] F. Prendin, S. Del Favero, M. Vettoretti, G. Sparacino, A. Facchinetti, Forecasting of Glucose Levels and Hypoglycemic Events: Head-to-Head Comparison of Linear and Nonlinear Data-Driven Algorithms Based on Continuous Glucose Monitoring Data Only, Sensors 21 (5) (2021) 1647.

[41] G. Sparacino, F. Zanderigo, S. Corazza, A. Maran, A. Facchinetti, C. Cobelli, Glucose concentration can be predicted ahead in time from continuous glucose monitoring sensor time-series, IEEE Transactions on biomedical engineering 54 (5) (2007) 931–937.

[42] C. Toffanin, F. Iacono, L. Magni, Personalized LSTM-based alarm systems for hypoglycemia prevention, 2023 Mediterranean Conference on Control and Automation (2023 - submitted).
URL `http://sisdin.unipv.it/lab/Main.pdf`

[43] C. Toffanin, S. Del Favero, E. Aiello, M. Messori, C. Cobelli, L. Magni, Glucose-insulin model identified in free-living conditions for hypoglycaemia prevention, Journal of Process Control 64 (2018) 27–36.

[44] C. Toffanin, E. M. Aiello, C. Cobelli, L. Magni, Hypoglycemia prevention via personalized glucose-insulin models identified in free-living conditions, Journal of Diabetes Science and Technology 13 (6) (2019) 1008–1016.

[45] T. Saito, M. Rehmsmeier, The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets, PloS one 10 (3) (2015) e0118432.

[46] C. Cobelli, C. Dalla Man, G. Sparacino, L. Magni, G. De Nicolao, B. P. Kovatchev, Diabetes: models, signals, and control, IEEE reviews in biomedical engineering 2 (2009) 54–96.

[47] C. Dalla Man, R. A. Rizza, C. Cobelli, Meal simulation model of the glucose-insulin system, IEEE Transactions on biomedical engineering 54 (10) (2007) 1740–1749.

[48] C. Dalla Man, F. Micheletto, D. Lv, M. Breton, B. Kovatchev, C. Cobelli, The UVA/PADOVA type 1 diabetes simulator: new features, Journal of diabetes science and technology 8 (1) (2014) 26–34.

[49] C. Toffanin, R. Visentin, M. Messori, F. Di Palma, L. Magni, C. Cobelli, Toward a run-to-run adaptive artificial pancreas: In silico results, IEEE Transactions on Biomedical Engineering 65 (3) (2017) 479–488.

[50] Y. M. Luijf, J. H. DeVries, K. Zwinderman, L. Leelarathna, M. Nodale, K. Caldwell, K. Kumareswaran, D. Elleri, J. M. Allen, M. E. Wilinska, et al., Day and night closed-loop control in adults with type 1 diabetes: a comparison of two closed-loop algorithms driving continuous subcutaneous insulin infusion versus patient self-management, Diabetes care 36 (12) (2013) 3882–3887.

[51] L. Magni, D. M. Raimondo, L. Bossi, C. Dalla Man, G. De Nicolao, B. Kovatchev, C. Cobelli, Model predictive control of type 1 diabetes: an in silico trial, Journal of Diabetes Science and Technology 1 (6) (2007) 804–812.

[52] P. Soru, G. De Nicolao, C. Toffanin, C. Dalla Man, C. Cobelli, L. Magni, A. H. Consortium, et al., Mpc based artificial pancreas: strategies for individualization and meal compensation, Annual Reviews in Control 36 (1) (2012) 118–128.

[53] J. Kropff, S. Del Favero, J. Place, C. Toffanin, R. Visentin, M. Monaro, M. Messori, F. Di Palma, G. Lanzola, A. Farret, et al., 2 month evening and night closed-loop glucose control in patients with type 1 diabetes under free-living conditions: a randomised crossover trial, The lancet Diabetes & endocrinology 3 (12) (2015) 939–947.

[54] E. Renard, A. Farret, J. Kropff, D. Bruttomesso, M. Messori, J. Place, R. Visentin, R. Calore, C. Toffanin, F. Di Palma, et al., Day-and-night closed-loop glucose control in patients with type 1 diabetes under free-living conditions: results of a single-arm 1-month experience compared with a previously reported feasibility study of evening and night at home, Diabetes Care 39 (7) (2016) 1151–1160.

[55] M. Messori, J. Kropff, S. Del Favero, J. Place, R. Visentin, R. Calore, C. Toffanin, F. Di Palma, G. Lanzola, A. Farret, et al., Individually adaptive artificial pancreas in subjects with type 1 diabetes: a one-month proof-of-concept trial in free-living conditions, Diabetes Technology & Therapeutics 19 (10) (2017) 560–571.