Investigating Learning Methodologies on Edge Devices for Blood Glucose Level Forecasting in Type 1 Diabetes Patients Using CGM Sensor Data

Federico D'Antoni [1], Paolo Giaccone [2], Lorenzo Petrosino [3], Tamara Boscarino [4], Anna Sabatini [5], Luca Vollero [6], Vincenzo Piemonte [7], Mario Merone [8]

[1, 2, 3, 5, 6, 8] Research Unit of Computer Systems and Bioinformatics,
Department of Engineering, Università Campus Bio-Medico di Roma, via Alvaro del Portillo, Rome, Italy

[4] Unit of Intelligent Health Technologies, Sustainable Design Management and Assessment,
Department of Engineering, Università Campus Bio-Medico di Roma, via Alvaro del Portillo, Rome, Italy

[7] Chemical-Physics Fundamentals in Chemical Engineering, Department of Engineering,
Università Campus Bio-Medico di Roma, via Alvaro del Portillo, Rome, Italy

Email: [1] f.dantoni@unicampus.it, [2] paolo.giaccone@unicampus.it,
[3] lorenzo.petrosino@unicampus.it, [4] tamara.boscarino@unicampus.it,
[5] a.sabatini@unicampus.it, [6] l.vollero@unicampus.it, [7] v.piemonte@unicampus.it,
[8] m.merone@unicampus.it

Abstract

Type 1 Diabetes mellitus (T1D) is a widespread disease characterized by a persistent condition of hyperglycemia. Continuous Glucose Monitoring (CGM) devices allow people with T1D to keep track of their glycemic level for 24 hours a day. Artificial intelligence models can aid people with T1D adjusting and optimizing their insulin therapy by providing a prediction of the future glycemic level based on CGM data; nonetheless, most of them are large models that run on the cloud, whereas few studies have focused on the application on an edge device. Applying a data-driven model that must be continuously updated on an edge-computing system requires a compromise between the predictive model performance and the limited computational capability of the edge device. In this study, we investigate different training approaches of a well-established Long Short-Term Memory neural network for blood glucose level forecasting in people with T1D based on CGM and insulin data. The best performance is achieved when the model is pre-trained on a large amount of data from 10 virtual patients, and fine-tuned on patient-specific data updating only the parameters of the output layer, while keeping the parameters of the hidden layers unchanged. The numeric results are comparable to those achieved by larger models in the literature. The presented model is characterized by an average training and DRQ time of 67.6 seconds on an edge device that is largely acceptable in practical cases.

Index Terms - Time series forecasting, Neural Networks, Edge Computing, Diabetes.

1. Introduction

Type 1 Diabetes mellitus (T1D) is a chronic autoimmune disorder characterized by the body's inability to produce insulin, resulting in elevated blood glucose levels [1]. Ac- curate forecasting of blood glucose levels plays a vital role in the effective management of T1D, as it allows patients to proactively adjust their insulin dosage, dietary intake, and physical activity, to maintain optimal glycemic control. Continuous Glucose Monitoring (CGM) systems have revolutionized diabetes care by providing real-time and historical blood glucose data. However, extracting meaningful insights from this data and generating accurate predictions remains a challenging task [2].

Traditional approaches for blood glucose level fore- casting primarily rely on statistical modeling techniques [3]. While these methods have shown some success, they often fail to capture the complex and dynamic nature of glucose metabolism, leading to limited accuracy and reliability in prediction [4]. As machine learning techniques continue to advance, there is a growing interest in exploring their potential to improve glucose level forecasting. Models for regression including Forests of Trees [5] and kernel machines[6] have achieved promising results in forecasting future CGM levels. However, the research is moving towards the utilization of deep learning models, including auto-regressive [7], Long Short-Term Memory (LSTM) [8], and Convolutional [9] neural networks. These models are able to learn the complex time dependence between the input and the output of the time series, and are thus particularly suited for such a task.

The advent of edge computing brings promising opportunities for enhancing the performance and efficiency of machine learning algorithms in the context of diabetes management [10]. Edge devices, such as small computing devices or wearable sensors, have the ability to process data locally, reducing latency and dependency on cloud- based infrastructure. Moreover, they do not rely on a continuous cloud connection for working, reducing the number of possible malfunctioning by processing data as close as possible to where they are collected [11]. This capability opens up new avenues for investigating different learning methodologies directly on the edge device, where the data is collected and analyzed in real-time. As mentioned, recent studies in time series tasks related to T1D employ deep learning algorithms, such as recurrent neural networks (RNNs) relationships in data makes them well-suited for modeling blood glucose dynamics. The predictive ability of such models depends on the depth of the neural network and the amount of data available for training. However, deploying these complex models on resource-constrained edge devices poses several challenges, including limited computational power, memory constraints, and energy efficiency considerations [13]; moreover, the accurate blood glucose level prediction with a precision-medicine approach may require dozens of recorded CGM days from a specific patient [14], which are usually not available in clinical practice.

For the reasons above, in this paper we aim to expand our previous study [14] investigating different learning methodologies specifically tailored for edge devices in the context of blood glucose level forecasting using CGM sensor data in patients with T1D, and validating the results on data from real patients and in a realistic data-gathering scenario. We evaluate different training approaches of a well-established model for blood glucose levels forecasting to address the aforementioned challenges. Our objective is to develop accurate and efficient forecasting models that can be deployed directly on edge devices, empowering individuals with T1D to make informed decisions regarding their daily management and treatment strategies. Finally, we investigate the amount of time required to train or update these models directly on the edge devices with the latest available data. Since the data taken as input by the model concern only information recorded and gathered by the CGM sensor and the insulin pump, the system is completely automated and does not require patient intervention to provide data manually. The advancement of personalized T1D care introduced by this study could enable the development of real-time, low-power, and reliable blood glucose forecasting systems that seamlessly integrate with patients' everyday lives, ultimately improving their quality of life and reducing the risk of complications associated with poor glycemic control.

2. Materials And Methods

To validate this study, we exploited data from either virtual and real patients. Data from virtual patients were used to train a preliminary configuration of the predictor, whereas real data were used to fine-tune the model and to evaluate the performance.

A. Virtual dataset

The UVA/Padova T1DM simulator [15] is used to generate data of 10 adult subjects with T1D. For each, 30 days of data are generated with 1-minute sampling. During each day, patients can randomly have from a minimum of 3 to a maximum of 5 meals, each having a reference carbohydrate intake taken from the Dietary Reference Intakes for Carbohydrate [16]. Breakfast, lunch, and dinner are always present and set at reference times 8:00, 13:00, and 20:00, with reference carbohydrate intake of 45, 70, and 80 grams, respectively. One or two snacks at 11:00 and/or 17:00 are randomly included with reference carbohydrate intake of 20 grams. To make data more realistic, the time of each meal is varied from the reference time by a random amount taken from a uniform discrete distribution in the interval [-60, +60] minutes, whereas the amount of carbohydrates for each meal is obtained by varying the reference by a casual value extracted from a uniform discrete distribution in the interval [-20, +20] grams. Hypoglycemic and hyperglycemic events are intentionally generated to match the glycemic excursions observed in real data; this is achieved by modifying the optimal insulin bolus automatically computed by the simulator itself in occurrence of each meal, in order to simulate human error on carbohydrate counting [17]. The corrective value is randomly taken in the discrete interval [-3, +3] units of insulin.

B. Real dataset

The Unit of Endocrinology and Diabetology of±Campus Bio-Medico University Hospital provided data of 12 people with T1D who used the M±edtronic Enlite CGM sensor coupled with an insulin pump, whose data were exploited in our previous study [18]. Data are± gathered with a 5-minute sampling. The dataset includes 7 females and 5 males, aged between 24 and 69 (average 40 ± 15) diagnosed with T1D from 1 to 40 years ago (average 16.2 ± 12.9). Each patient was monitored for a period of time ranging from 5 to 25 days (average 10.6 ± 5.2), for a total of 127 days of monitoring; the longest monitoring period without data interruption for any patient is 5 days.

C. Data Preprocessing

In order to ensure consistency between the datasets considered in this study, data generated with the UVA/Padova simulator are modified according to the features of the real data. Straightforwardly, they were up-sampled taking one value every 5 minutes and no further feature was considered beside CGM and insulin; this means that the model does not exploit any feature that must be manually provided by patients such as carbohydrate intake, resulting completely automated. The insulin feature was represented through the Insulin-On-Board (IOB), a combination of basal and bolus insulin which represents an estimation of the amount of insulin that is active in the subject’s body after a determined time from the injection. The active insulin time is equal to 3 hours and its action is linear [19] and computed for each timestamp *t* as:

$$\sum_{k=0}^{179} a(k)I(t-k) \qquad (1)$$

where $I(t-k)$ represents the value of total insulin injected at minute $(t-k)$, $a(k) = (180-k)/180$ is the coefficient corresponding to the time-discrete insulin decay curve according to a 1-minute timestamp of insulin delivery, and k = 0, 1, 2, ..., 179 are the total timestamps for the 3 hours of active insulin.

Separately for each patient in the virtual or the real dataset, a Z-score standardization is applied to the features to obtain a normal distribution of the values with zero mean and unit standard deviation. Finally, the dataset is split into windows consisting of 30 minutes of data (input of the model) and one sample as label (output of the model). A prediction horizon (PH) is fixed to determine how forward in time the prediction is performed. Thus, at each timestamp *t*, the estimated future

CGM value $\widehat{\mathrm{CGM}}$ is a function of CGM and IOB values

from the previous 30 minutes:

$$\widehat{CGM}(t+PH) = f(CGM[t-30,\dots,t]; IOB[t-29,\dots,t]) \qquad (2)$$

where the PH is set to 30 minutes which is the most widely adopted value in the literature, as it would allow sufficient time to prevent an adverse event [20].

D. Neural network architecture

An LSTM architecture is selected as forecasting model, i.e., a specific RNN variant that is very suitable to handle long-term dependencies [21]. An LSTM is an RNN where any cell at a considered timestamp *t* contains an internal memory vector, or state vector, c_t that defines its state as described in the following. The units of the network are composed of an input cell, an output cell and a forget gate. Considering the matrices **W** input weights, **R** recurrent weights and **b** bias for each gate, the state of the cell at timestamp *t* defined as

$$\mathbf{c_t} = \mathbf{f_t} \odot \mathbf{c_{t-1}} + \mathbf{i_t} \odot \mathbf{g_t} \qquad (3)$$

whereas the hidden state is defined by

$$\mathbf{h_t} = \mathbf{o_t} \odot \sigma(\mathbf{c_t}) \qquad (4)$$

where the terms, for a given input $\mathbf{x}(t)$, are referred to the equations governing the gates. In detail, for the input gate:

$$\mathbf{i_t} = \sigma(\mathbf{W_{in}}\mathbf{x}(t) + \mathbf{R_i}\mathbf{h_{t-1}} + \mathbf{b_i}) \qquad (5)$$

for the forget gate:

$$\mathbf{f_t} = \sigma(\mathbf{W_f}\,\mathbf{x}(t) + \mathbf{R_f}\,\mathbf{h_{t-1}} + \mathbf{b_f}\,) \qquad (6)$$

for the cell candidate:

$$\mathbf{g_t} = \sigma(\mathbf{W_g x}(t) + \mathbf{R_g h_{t-1}} + \mathbf{b_g}) \tag{7}$$

and, finally, for the output gate:

$$\mathbf{o_t} = \sigma(\mathbf{W_o x}(t) + \mathbf{R_o h_{t-1}} + \mathbf{b_o}) \tag{8}$$

where σ is the sigmoid activation function.

The optimal hyperparameters of the network were identified in our previous work through the medium of a grid search [14], resulting in a 3-layered network composed, in sequence, of an LSTM layer with 16 cells, a dense layer with 10 neurons, and a single output neuron for regression. A learning rate of 0.01, Adam as optimizer, and Rectified Linear Unit (ReLU) activation function for the dense and output layers also resulted from the optimization. It is worth noting that the proposed model has a relatively limited number of layers and parameters and is thus particularly light weighted and suited for running on the edge.

E. Experimental Setup

The main contribution of this paper consists in the investigation of different learning strategies of a model for glucose levels forecasting on an edge device. Indeed, previous studies showed that following a precision-medicine approach, i.e., the utilization of a patient-specific model that is trained exclusively on data from the same patient that will be used for testing, usually achieves the best performance [22, 23]. Nonetheless, such an approach often results in suboptimal training of a deep learning model due to the limited amount of subject-specific data that are available in practical applications. For this reason, other approaches enlarge the training dataset including data from several patients to improve training performance. Following these considerations, the LSTM neural network model described in the previous section has been evaluated according to the following approaches:

- Transfer Learning (TL): data from all the virtual patients are put together to compose a whole dataset for the training of the neural network; then, the trained model is used in a test-only setup on data of the real patients.
- Fine Tuning (FT): the model is initialized using the weights obtained with the TL approach; then, for each testing patient, the first available data are used to retrain the model following a precision medicine approach.
- Fine Tuning with frozen layer 1 (FL1): same as the FT approach, but the weights of the first layer (LSTM layer) are not set to trainable, so they remain those resulted from the training on the virtual subjects.
- Fine Tuning with frozen layer 2 (FL2): the weights of the second layer (dense layer) are not set to trainable.
- Fine Tuning with frozen layers 1 and 2 (FL12): the weights of both the LSTM and the dense layer are not set to trainable, thus the only training concerns the parameters of the output neuron; this is the configuration with the least number of trainable parameters.
- Precision Medicine (PM) only: the model is trained from scratch on patient-specific data, without pre- training on the virtual dataset. This approach requires all the parameters to be trained from a random initialization and has the most onerous training.

Since all real patients have at least 5 monitored days, data of each subject were split into a discovery set (days 1 to 4 included) and a test set (days 5 to last recorded day). In this way, the test set is fixed, allowing a fair comparison between tests with a different number of training days. It is worth noting that the amount of training days was increased backwards, i.e., day 4 was used for training when using 1 day of training, days 3 and 4 when training for 2 days, and so on, so that the discovery set for each test is composed of the days immediately preceding the test set.

The discovery set of each patient was subsequently split into a training and a validation set with a fixed 70/30% ratio in each experiment. To avoid overfitting, early-stopping was utilized to stop the training if the performance on the validation set did not improve for more than 6 consecutive epochs. All the training approaches were implemented in Python using the open-source libraries of TensorFlow and Keras; The optimization stage was performed utilizing the Google Colaboratory environment; the best-performing model was eventually run on the edge device in order to evaluate the training, quantization, and inference times.

F. Edge System

The edge system utilized in this study is the Raspberry Pi 4. It includes a quad core Cortex-A72 1.5GHz processor with 8GB of RAM; additional information can be found in the datasheet [24]. The steps followed for implementing and testing the model on edge architecture are as follows:

- The pre-trained model (as described in section E as TL), saved in the *h5* format, is loaded on the device.
- Fine-tuning is performed on the device to adapt the general model to a specific patient data.
- The trained model is then converted in the TensorFlow Lite format using concrete functions to allow a faster prediction [25]. This type of model quantization is called Dynamic Range Quantization (DRQ) and it provides reduced memory usage and faster computation without significantly affecting the model performance.
- Finally, to extract inference metrics, tests are con- ducted using the model in its TensorFlow Lite for mat.

3. Results And Discussion

The results are evaluated following the most widely utilized metrics for blood glucose levels forecasting, namely Root Mean Squared Error (RMSE), Mean Absolute Relative Difference (MARD), and Sum of Squares of Glucose Prediction Error (SSGPE), which is a metric specifically suited for evaluating glucose forecasting errors [26]. These metrics are based on the vector of errors between the predicted and real CGM values, defined as:

$$\boldsymbol{e} = \widehat{\boldsymbol{CGM}} - \boldsymbol{CGM} \qquad (9)$$

that allows the calculation of the metrics as:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{T} e(t)^2}{T}} \qquad (10)$$

$$MARD = \sqrt{\frac{\sum_{t=1}^{T} |e(t)|^2}{CGM(t)}} \qquad (11)$$

$$SSGPE = 100 \times \sqrt{\frac{\sum_{t=1}^{T} e(t)^2}{\sum_{t=1}^{N} CGM(t)^2}} \qquad (13)$$

where *t* is a generic timestamp and *T* is the total number of predicted timestamps. The RMSE provides a measure of the magnitude of the mean error, whereas MARD and SSGPE provide a percentage error that is independent of the magnitude of the observed quantity.

The average results for each training approach, for the different metrics, and for a number of training days varying from 1 to 4 are reported in Table I, and are graphically represented in Fig. 1. The results are in line with those reported in the literature, and considered acceptable to potentially avoid or mitigate an adverse glycemic event [2, 27]. However, it is worth stressing that the goal of this study is not to optimize the selected predictive model on the data, but rather to determine the optimal strategy for practical implementation on an edge device.

Table I: Average numeric results in terms of RMSE, MARD, and SSGPE for the different training approaches considered, for a number of training days varying from 1 to 4. The best results for each metric and for different training days are highlighted.

Metric	Training days	Training approach					
		PM	**FT**	**FL1**	**FL2**	**FL12**	**TL**
RMSE [mg/dL]	1	21.2	22.2	21.2	21.2	**21.0**	24.7
	2	**19.1**	21.2	19.2	**19.1**	19.4	-
	3	20.5	22.0	20.6	20.5	**20.4**	-
	4	20.4	20.9	20.4	**20.2**	**20.2**	-
MARD [%]	1	22.2	23.2	23.7	21.7	23.2	**17.4**
	2	**9.3**	10.1	**9.3**	9.4	9.5	-
	3	**10.2**	10.9	**10.2**	10.3	10.4	-
	4	10.8	11.0	10.9	**10.7**	10.8	-
SSGPE [%]	1	12.6	13.1	12.6	12.6	**12.5**	14.7
	2	**10.8**	11.9	**10.8**	**10.8**	10.9	-
	3	12.1	13.0	12.1	12.1	**12.0**	-
	4	**12.4**	12.8	12.5	**12.4**	**12.4**	-

It is worth noting that all the training approaches achieve similar performance, and there is not a specific approach that performs clearly better than the others. However, it is interesting to point out that the FT approach (i.e., re-training the whole model after having set the initial weight values) does not achieve the best results in any test, and is constantly outperformed by the fine-tuning approaches that consider one or more non-trainable layers. As one could expect, the TL approach achieves competitive results only when the others are trained for 1 day, as it is quickly outperformed by patient- suited models. Interestingly, the best results are achieved for a training time of 2 days; this could be due to the inclusion of very noisy data when increasing the discovery sets to 3 or 4 days.

Although by a small margin, the best approaches result to be the PM, achieving the best results in 5 tests, and the FL12, achieving the best results in 6 tests; however, the latter has the least amount of parameters to train, so it can be considered the most suited approach for training on an edge device.

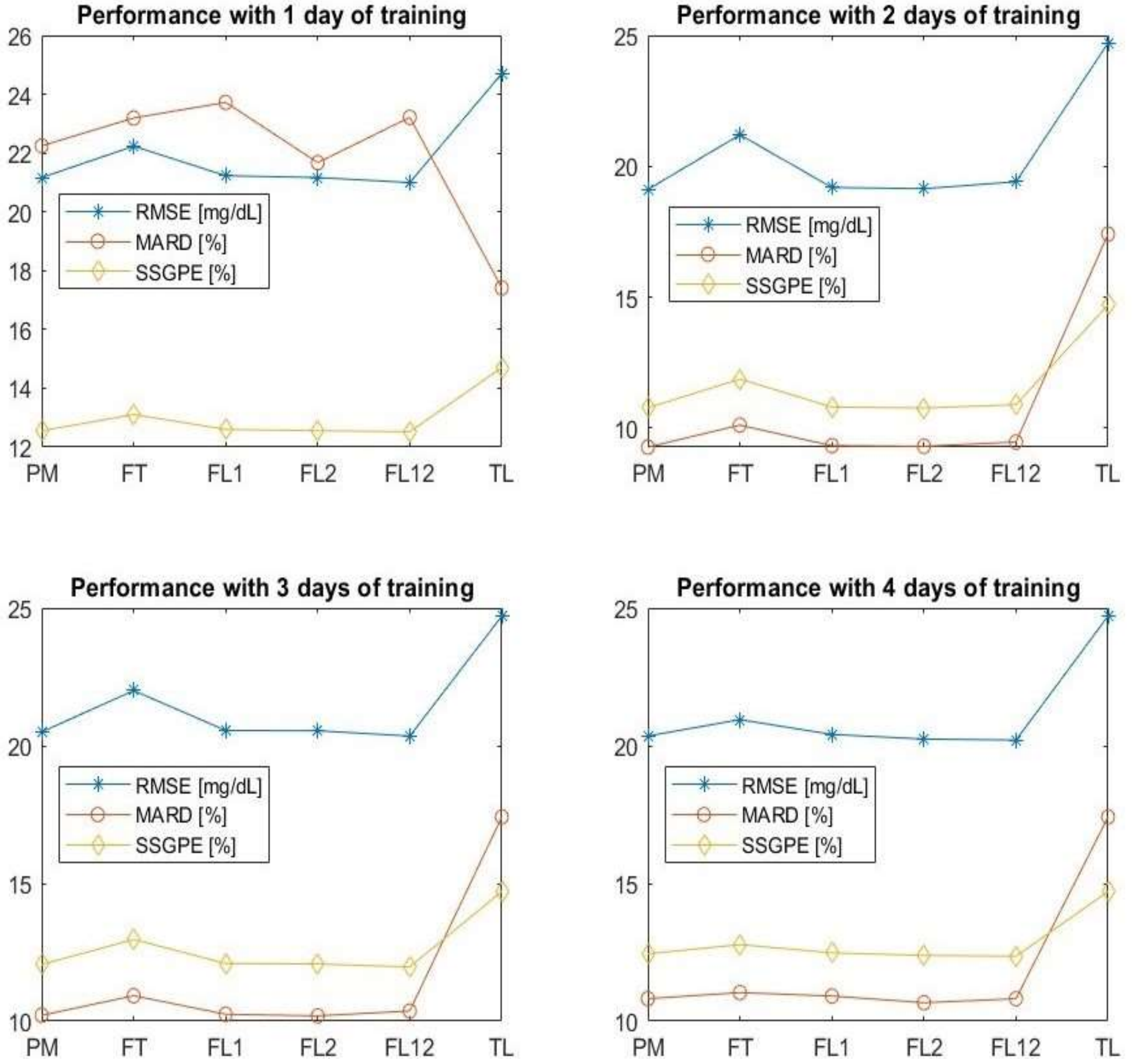

Figure 1: Results of the different configurations ranging from 1 to 4 days of training data.

Straightforwardly, we implemented this model on the edge device described in Sec. F in order to evaluate the real-life feasibility in terms of training, DRQ, and inference time. Table II reports the average (with standard deviation), maximum, and minimum times required for the tasks above. The edge implementation concerns only the models trained using 4 days of data, in order to evaluate the worst-case scenario in terms of time required, corresponding to the maximum size of the discovery set. The maximum inference time observed in any test is 1 millisecond and is thus negligible in this analysis. The average total times required for training on the edge and DRQ sum to 67.6 seconds, with a maximum time of 134.1 seconds. This is a suitable time for running on an edge device: considering the typical 5-minute sampling time of the CGM devices, corresponding to a prediction every 5 minutes, a model update on the edge device would be feasible in practice without affecting or limiting the predictions. This is also achieved due to the very limited number of trainable parameters of the FL12 approach. Moreover, such a small execution time is unlikely to considerably affect the battery charge of the edge device. This is important because a long training time could result in a complete battery discharge, resulting in the risk of leaving the patient without a decision support system.

Table II: Execution times on the edge device.

Time [*s*]	Training	DRQ	Inference
Average	51.2 ± 16.3	16.4 ± 8.9	$(6 \pm 3) \times 10^{-4}$
Minimum	19.1	12.0	3×10^{-4}
Maximum	93.7	40.4	1×10^{-3}

4. Conclusions

In this study, we investigated different training approaches of a well-established LSTM neural network for blood glucose level forecasting in people with T1D based on CGM and insulin data, without requiring the patient to provide manual information. The best performance is achieved by the model pre-trained on a large amount of data from virtual patients, and fine-tuned while updating only the parameters of the output layer, while keeping the parameters of the hidden layers frozen. This results in a small training time on an edge device that is largely acceptable in clinical practice. Future developments of this study could include the enrollment of a larger number of real patients, and the implementation of tests on different edge devices with smaller computational capacity.

References

[1] Michael J Fowler. Microvascular and macrovascular com- plications of diabetes. *Clinical diabetes*, 26(2):77–82, 2008.

[2] Virginie Felizardo, Nuno M Garcia, Nuno Pombo, and Imen Megdiche. Data-based algorithms and models using diabetics real data for blood glucose and hypoglycaemia prediction–a systematic literature review. *Artificial In- telligence in Medicine*, page 102120, 2021.

[3] Jaques Reifman, Srinivasan Rajaraman, Andrei Gribok, and W Kenneth Ward. Predictive monitoring for im- proved management of glucose levels. *Journal of Diabetes Science and Technology*, 1(4):478–486, 2007.

[4] Giovanni Sparacino, Andrea Facchinetti, Alberto Maran, and Claudio Cobelli. Continuous glucose monitoring time series and hypo/hyperglycemia prevention: require- ments, methods, open problems. *Current diabetes re- views*, 4(3):181–192, 2008.

[5] Eleni I Georga, Vasilios C Protopappas, Demosthenes Polyzos, and Dimitrios I Fotiadis. A predictive model of subcutaneous glucose concentration in type 1 diabetes based on random forests. In *Int. Conf. of the IEEE Eng. in Medicine and Biology Society*, pages 2889–2892, 2012.

[6] Takoua Hamdi, Jaouher Ben Ali, Véronique Di Costanzo, Farhat Fnaiech, Eric Moreau, and Jean-Marc Ginoux. Accurate prediction of continuous blood glucose based on support vector regression and differential evolution algorithm. *Biocybernetics and Biomedical Engineering*, 38(2):362–372, 2018.

[7] Federico D’Antoni, Mario Merone, Vincenzo Piemonte, Giulio Iannello, and Paolo Soda. Auto-regressive time delayed jump neural network for blood glucose levels fore- casting. *Knowledge-Based Systems*, page 106134, 2020.

[8] Maria Athanasiou, Konstantia Zarkogianni, Konstanti- nos Karytsas, and Konstantina S Nikita. An lstm-based approach towards automated meal detection from continuous glucose monitoring in type 1 diabetes mellitus. In *2021 IEEE 21st International Conference on Bioin- formatics and Bioengineering (BIBE)*, pages 1–5. IEEE, 2021.

[9] Touria El Idrissi and Ali Idri. Deep learning for blood glucose prediction: Cnn vs lstm. In *International Con- ference on Computational Science and Its Applications*, pages 379–393. Springer, 2020.

[10] William R Hersh, Mark Helfand, James Wallace, Dale Kraemer, Patricia Patterson, Susan Shapiro, and Mer- wyn Greenlick. Clinical outcomes resulting from telemedicine interventions: a systematic review. *BMC Medical Informatics and Decision Making*, 1(1):1–8, 2001.

[11] Federico D’Antoni, Lorenzo Petrosino, Fabiola Sgarro, Antonio Pagano, Luca Vollero, Vincenzo Piemonte, and Mario Merone. Prediction of glucose concentration in children with type 1 diabetes using neural networks: An edge computing application. *Bioengineering*, 9(5):183, 2022.

[12] Youru Li, Zhenfeng Zhu, Deqiang Kong, Hua Han, and Yao Zhao. EA-LSTM: Evolutionary attention-based LSTM for time series prediction. *Knowledge-Based Systems*, 2019.

[13] Taiyu Zhu, Lei Kuang, John Daniels, Pau Herrero, Kezhi Li, and Pantelis Georgiou. Iomt-enabled real-time blood glucose prediction with deep learning and edge computing. *IEEE Internet of Things Journal*, 2022.

[14] Federico D’Antoni, Lorenzo Petrosino, Andrea Velieri, Daniele Sasso, Onorato D’Angelis, Tamara Boscarino, Luca Vollero, Mario Merone, and Vincenzo Piemonte. Identification of optimal training for prediction of glu- cose levels in type-1-diabetes using edge computing. In *2022 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICEC- CME)*, pages 1–5. IEEE, 2022.

[15] Chiara Dalla Man, Francesco Micheletto, Dayu Lv, Marc Breton, Boris Kovatchev, and Claudio Cobelli. The UVA/PADOVA type 1 diabetes simulator: new features. *Journal of diabetes science and technology*, 8(1):26–34, 2014.

[16] Joanne R Lupton, JA Brooks, NF Butte, B Caballero, JP Flatt, SK Fried, et al. Dietary reference intakes for energy, carbohydrate, fiber, fat, fatty acids, choles- terol, protein, and amino acids. *National Academy Press: Washington, DC, USA*, 5:589–768, 2002.

[17] Sina Buck, Collin Krauss, Delia Waldenmaier, Christina Liebing, Nina Jendrike, Josef Högel, Boris M Pfeiffer, Cornelia Haug, and Guido Freckmann. Evaluation of meal carbohydrate counting errors in patients with type 1 diabetes. *Experimental and Clinical Endocrinology & Diabetes*, 130(07):475–483, 2022.

[18] Federico D'Antoni, Mario Merone, Vincenzo Piemonte, Paolo Pozzilli, Giulio Iannello, and Paolo Soda. Early experience in forecasting blood glucose levels using a de- layed and auto-regressive jump neural network. In *2019 IEEE 18th Int. Conf. on Cognitive Informatics & Cog- nitive Computing (ICCI*CC19)*, pages 394–402. IEEE, 2019.

[19] Howard Zisser, Lauren Robinson, Wendy Bevier, Eyal Dassau, Christian Ellingsen, Francis J Doyle III, and Lois Jovanovic. Bolus calculator: a review of four "smart" insulin pumps. *Diabetes technology & therapeu- tics*, 10(6):441–444, 2008.

[20] Omer Mujahid, Ivan Contreras, and Josep Vehi. Machine learning techniques for hypoglycemia prediction: Trends and challenges. *Sensors*, 21(2):546, 2021.

[21] Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. A review of recurrent neural networks: LSTM cells and network architectures. *Neural computation*, 31(7):1235–1270, 2019.

[22] Silvia Oviedo, Josep Vehí, Remei Calm, and Joaquim Ar- mengol. A review of personalized blood glucose predic- tion strategies for T1DM patients. *International jour- nal for numerical methods in biomedical engineering*, 33(6):e2833, 2017.

[23] Eleni I Georga, Dimitrios I Fotiadis, and Stelios K Tigas. *Personalized Predictive Modeling in Type 1 Diabetes*. Academic Press, 2017.

[24] Copyright 2019 Raspberry Pi (Trading) Ltd. All rights reserved. Link to the official datasheet:. *https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf*, 2019.

[25] Tensorflow. Link to the official guide:. *https://www.tensorflow.org/lite/guide*, Last updated 2022-06-28.

[26] Matteo Gadaleta, Andrea Facchinetti, Enrico Grisan, and Michele Rossi. Prediction of adverse glycemic events from continuous glucose monitoring signal. *IEEE Jour- nal of Biomedical and Health Informatics*, 23(2):650–659, 2018.

[27] Omar Diouri, Monika Cigler, Martina Vettoretti, Julia K Mader, Pratik Choudhary, Eric Renard, and HYPO- RESOLVE Consortium. Hypoglycaemia detection and prediction techniques: A systematic review on the lat- est developments. *Diabetes/Metabolism Research and Re- views*, 37(7):e3449, 2021.