International Advance Journal of Engineering Research (IAJER)
Volume 4, Issue 10 (October- 2021), PP 01-08
ISSN: 2360-819X
www.iajer.com

Research Paper **Open Access**

BLOOD GLUCOSE LEVEL PREDICTION USING DEEP LEARNING NETWORKS

Anel Hasić[1], Tatjana Konjić[1] , Amira Šerifović-Trbalić[1]
[1](*Faculty of Electrical Engineering, University of Tuzla, Bosnia and Herzegovina)*
**Corresponding author: Anel Hasić[1]*

ABSTRACT: *Blood glucose prediction and management is essential for preventivediabetes management. The continuous glucose monitoring (CGM) system represents a valuabletool for closely monitoring and reacting to patient'scurrentblood glucose levels and trends, and thusithelps patients to better manage theirinsulin distribution. The data obtained by the CGM system can beused to train machine learningmodels to predict future blood glucose levels. In thispaper, wepresent an approachbased on recurrentdeeplearningmodelstrained in an end-to-end fashion to predictblood glucose levelsfrom CGM measurements***.**

Keywords *-Blood glucose level, CGM, deep learning networks, LSTM, stacked LSTM*

I. INTRODUCTION

Diabetes mellitus represents a chronic metabolic disorder that results in an abnormal blood glucose level that might lead to a considerable complication for a human system, and it can significantly reduce the quality of life, as well as the life expectancy of a patient. According to the research published in [1], patients with diabetes have an increased risk of morbidity and mortality compared to a healthy person.

With an increasing number of people with diabetes, diabetes is considered as severe chronic disease that requires long-term management of blood glucose levels. The standard approach for controlling diabetes is to measure the blood glucose (BG) levels of the patient several times a day, by taking a blood sample from the fingertip. The recent advances in technology have led to the design of devices for the continuous measurement of blood glucose levels (CGM). Such devices can measure blood glucose levels every 5 minutes [2].

Further improvement in the process of controlling blood glucose levels is possible with the help of prediction. Blood glucose prediction is the process of predicting blood glucose levels by taking into account the patient's previous and current condition, mainly to avoid complications caused by hyperglycaemia or hypoglycaemia. Many factors can affect blood glucose levels, and some of them are: a carbohydrate intake, a recent insulin injection, physical activity, diet, stress levels, sleeping patterns, medications, the presence of an infection in the body, allergies, altitude and others. The ideal predictor of blood glucose levels should incorporate as much information as possible to effectively predict glucose levels. This makes predicting the short-term blood glucose changes (up to a few hours) a challenging task, and development of machine learning approaches imposes as an obvious approach for improving patient care. So far, various statistical and machine learning methods have been proposed for blood glucose prediction and they mainly focus on the short-term predictions based on the obtained values from the CGM devices or in combination with the information about the insulin dose and/or meal intake.

Early research in the field of blood glucose prediction was focused on the use of autoregressive models [3, 4]. For example, in [5], the classical statistical method, i.e., autoregressive integrated moving average (ARIMA) model-based algorithm, was used for blood glucose prediction in 30 to 60 minutes prediction horizons. The use of the support vector regression (SVR) in predicting glucose up to 120 minutes prediction horizons for type 1 diabetes was presented in [6], while in [7], a generic physiological model of blood glucose dynamics was used to generate informative features for a SVR model trained on patient specific data.

Artificial neural networks (ANN) and neuro-fuzzy systems were proposed for prediction of the blood glucose level in several research [8-10]. In [9], for example, a feed-forward neural network is designed with

eleven neurons in the input layer (corresponding to variables such as CGM data, the rate of change of glucose levels, meal intake and insulin dosage), and nine neurons with hyperbolic tangent transfer function in the hidden layer. The network was trained with the use of data from 17 patients and tested on data from 10 other patients for a 75-min prediction horizon. Recurrent Neural Networks (RNNs) have shown its capability in prediction of glucose levels [8], but the major challenges in designing such systems using classical RNNs is their limited capacity to learn long-term dependencies, because of the vanishing or exploding gradient problem.

Deep learning methods have shown superior performance compared to traditional machine learning. This is due to their ability to automatically extract and learn relevant features with higher complexity and representations from the training samples in order to accurately predict the blood glucose. For example, in [11], a convolutional neural network (CNN) model was proposed to predict future blood glucose levels. The model was a modified version of WaveNet. Within the deep learning framework, RNNs are typically implemented with Long Short-Term Memory (LSTM) units that provide a gating mechanism enabling the RNNs to preserve in memory events further in the past [12-14], but there are some researches, for example in [15], where the convolutional RNNs are investigated for BG prediction.

In this paper, we focus on using a single layer LSTM and stacked LSTM deep learning neural network to predict blood glucose levels based on CGM measurements. Different prediction horizons observed. Developed models prediction accuracy evaluated by the root mean square error and Pearson correlation coefficient. It was confirmed that stacked LSTM network gave better results than single layer LSTM in each prediction horizon.

The remainder of this paper is structured as follows. The variations of LSTM are introduced in Section 1. The dataset and training process are described in Section 2, while the evaluation and results are presented in Section 3, and finally conclusions in Section 4.

II. LSTM NETWORK

Traditional RNNs can easily learn short-term dependencies, but they have difficulties in learning long-term dependencies due to the problem of vanishing and exploding gradients. The Long Short-Term Memory (LSTM) network is a type of RNN [16] widely used for various sequence prediction and sequence labelling tasks, because of its superior performance in accurate modelling of both short-term and long-term data dependencies.

LSTMs are designed to avoid the problem of long-term dependencies. Remembering of the information for a long period of time is practically their default behaviour. If we compare LSTM networks with RNN networks, we can see that LSTM, same as other recurrent neural networks, have the shape of a chain of repeating modules of a neural network. In standard RNNs, this repetition module will have a very simple structure, such as a single tanh layer.

LSTMs consist of memory cells with a cell state that represents a memory of the cell and maintains the information through time, and a gate structure that controls the flow of information. The structure of the LSTM cell is shown in Fig. 1.

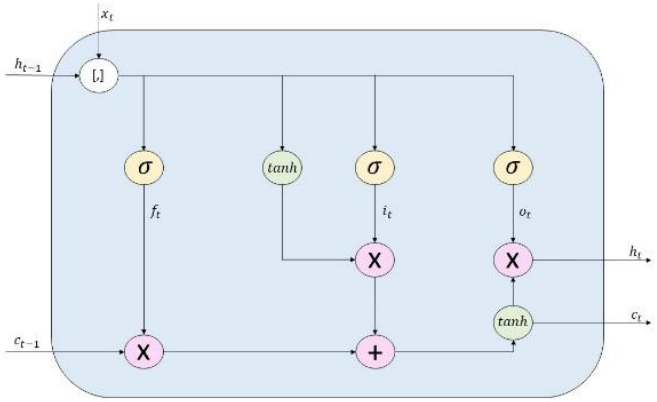


Fig. 1LSTM cell

The state of the cell passes through theentire chain, with only some minor interactions, which allows the information to pass unchanged through the structure. Each arrow carries an entire vector, from the output of one node to the input of another node. Pink circles represent operations at points, such as vector addition, while yellow frames represent learned layers of the neural network. Merging arrows indicate joining, while separating arrows indicate copying the contents of one node to another node.

An LSTM is an RNN where the cell at each step t contains an internal memory vector c_t, and three gates controlling what parts of the internal memory will be kept (the forget gate f_t,), what parts of the input that will be stored in the internal memory (the input gate i_t,), as well as what will be included in the output (the output gate o_t,). A σ and $tanh$ are the activation functions, the former is the sigmoid function and the latter is the hyperbolic tangent function. LSTM protects and controls information in cells through gates. Each gate has its own parameters, biases and weights. The inputs to the gates are the input vector, $x_t$and the previous outputs (hidden states) h_{t-1},. In order to compute the new internal memory ct and the cell output h_t, at each step, following expressions are evaluated.

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{1}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{2}$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{3}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{4}$$

$$h_t = o_t \cdot tanh(c_t) \tag{5}$$

The matrices W_f, W_i, W_c, and W_o are the weight matrices that are applied to the LSTM cell input, while U_f, U_i, U_c i U_o are the previous output weight matrices, i.e., the peephole weights, and b_f, b_i, b_c and b_o are the biases (bias vectors). Each gate in the cell has a specific and unique function. The forget gate f_t determines which information can be carried to the cell from the previous output of the LSTM cell. Input gate it acts on the previous state h_{t-1}, after being modified with forget gate, and decides how much the new state h_t should be updated with the new candidate. To produce the result h_t, the cell first filters its current state with nonlinear tanh function. After that, the output gate o_t selects the part of the state to be returned as the output. Each gate depends on the current external input x_t and the output of the previous cells h_{t-1}.

From the previous expressions it is possible to notice that in LSTM the issue of the vanishing gradient does not occur, due to adding forget, input and output gates which produce paths where gradients can flow more constantly in longer-term without vanishing.

In practice, the update and forget gates are never fully open or closed due to the functional shape of the sigmoid, which saturates only for infinite values. As a result, even if long-term memory in LSTM is significantly improved compared to the traditional RNN architectures, the content of the cell cannot be kept completely unchanged over time.

As a potential approach to improve the LSTM network performance, one can try to increase the depth of the model architecture by employing the stacking technique [17]. The structure of the stacked LSTM network is shown in Fig. 2.

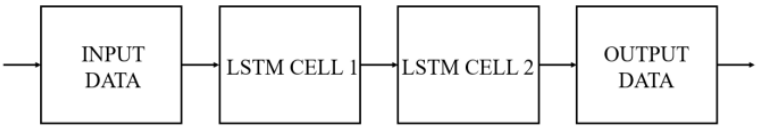


Fig. 2Stacked LSTM network structure

LSTM networks are trained by the Backpropagation through time algorithm (BPTT) [18]. In order to complete the neural network algorithm, it is necessary to calculate the gradient of the error function. An algorithm known as Error backpropagation (Backprop), devised in the 1970s, was first introduced into neural networks in 1986 and has remained the most common form of calculating the loss function gradient ever since. In the network learning process, it is initially necessary to determine the derivatives of the error function in relation to the weights and thresholds, and using the backward error forwarding algorithm, this procedure can be easily implemented in computer programs. In the second phase, these derivatives are used to adjust the weights.

Gradient-based learning requires a closed-form relationship between model parameters and loss function. This relationship allows the spread of the gradient data calculated on the loss function to the model parameters, in order to change them accordingly. Although this operation is simple on models represented by a directional acyclic graph, such as an advanced neural network (FFNN), care should be taken when this is applied to RNNs, whose corresponding graph is cyclic. In order to find a direct relationship between the loss function and the network weight, the RNN must be represented as an equivalent infinite, acyclic, and directed graph. The procedure is called unfolding and consists of copying the hidden layer network for each time interval, obtaining a certain type of FFNN.

The main difference of extended RNN compared to standard FFNN is that the weight matrices are limited to accepting the same values in all layer replicas, as they represent a recursive application of the same operation. Through this transformation, the network can be enabled by standard learning algorithms, originally conceived for advanced architectures.

This learning process is called Backpropagation Through time, and is one of the most successful techniques adopted for RNN training. Although the network structure may initially be repeated indefinitely several times, in practice the unfolding is always shortened after a certain time.

In this learning process called abbreviated BPPT, the folded architecture is repeated up to a certain number of steps τ_b with the upper limit of the length of the time series.

III. EXPERIMENTAL SETUPS

In this paper, we consider the problem of predicting future blood glucose levels using only previous blood glucose level measurements obtained by CGM devices provided in [19].

A CGM device was used to collect data, at 5-min intervals, for each of the patients. Datasets used for model construction consist of 10000 samples, which are partitioned into training set to 80%, while the leftover data sets are used as the testing set.There are the same number of blood glucose level observations for each patient in the training and testing datasets. All input data to the LSTM have been normalized in the range of 0–1 to prevent unforeseen results and to facilitate the proper learning of the neural networks. Normalization was done by subtracting measured value from the mean value of the collected data, and dividing obtained result by the standard deviation of the data.When training and testing the LSTMs, only historical windows with no missing values are used.The training sliding window size is 45 minutes (9 steps) as it demonstrated better performance than other window sizes. The prediction horizons (PH) are 15, 30 and 45 minutes.

A number of hyper-parameters for the networks were explored and tuned to achieve the best results for our proposed models. The chosen hyper-parameters are: the batch size was set to 128, the number of training epochs was 250, LSTM hidden layer size was set to 200 for a single layer LSTM and 150 for a stacked LSTM. The initial learning rate for the stochastic Adam optimizer was set to 0.005.

We tested several stacked LSTM models, but the best performances are obtained with the model consisting of two LSTM layers. The first LSTM layer provides a sequence output that is fed as input to the second LSTM layer. Both LSTM layers have the same internal architecture described earlier. The second LSTM layer was followed by a fully connected layer with a sigmoid activation function to predict BGL values.

IV. RESULTS

To evaluate performance of the developed models, the root mean square error (RMSE) and the Pearson correlation coefficient (PCC) were used. RMSE indicates the difference between the actual data, t_i, and the predicted data, y_i, and it can be calculated as:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(t_i - y_i)^2} \tag{6}$$

where N is the number of predicted samples.

The second parameter used to evaluate models is Pearson correlation coefficient. This coefficient is

$$PCC = \frac{\sum_{i=1}^{N}(t_i - \bar{t})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(t_i - \bar{t})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}} \tag{7}$$

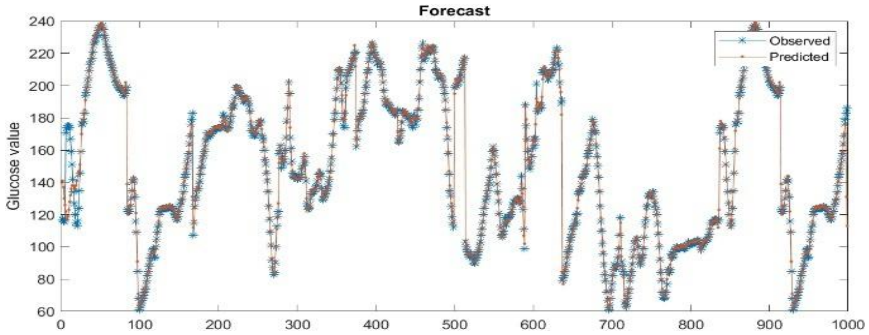
where $\bar{t}$ and $\bar{y}$are the mean values of the observed and predicted data, respectively.

The data from the output of the neural network were normalized to zero mean value and restored to their original form, in order to demonstrate the performance of the test data. Both LSTM networks, single layer LSTM and stacked LSTM, were used to predict the upcoming blood glucose levels for prediction horizons of 15, 30 and 45 minutes. The value of RMSE and PCC are summarized in Table 1. The stacked LSTM network outperforms the single layer LSTM network in all PHs with respect to the evaluation criteria

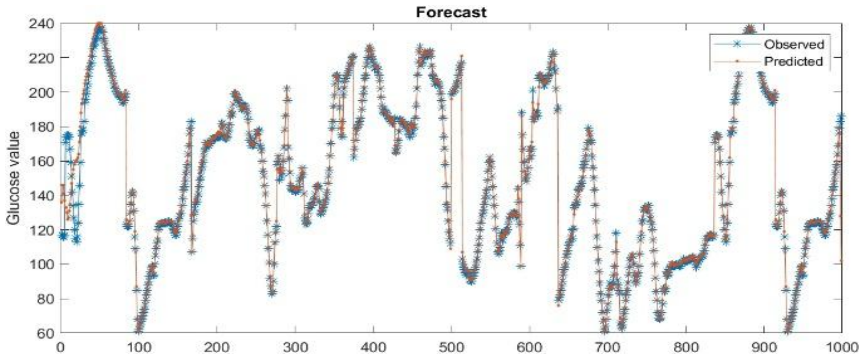
Table 1 RSME (mg/dl) and Pearson correlation coefficient

Prediction horizon	**Single LSTM**		**Stacked LSTMs**	
	RMSE	**PCC**	**RMSE**	**PCC**
15 min	3.9848	0.9909	3.5928	0.9921
30 min	10.9306	0.9746	6.8138	0.9872
45 min	12.5473	0.9659	8.926	0.9837

It is difficult to compare obtained RMSE results with other published researches due to differences in the used database and the observed prediction horizons. However, comparing the same networks (single layer and stacked LSTM) applied to blood glucose prediction presented in [20], [21], it is possible to notice that obtained RMSEs are even better. Changes of blood glucose during observed periods of 15, 30 and 45 minutes, were shown in Figures3,4 and 5, respectively.

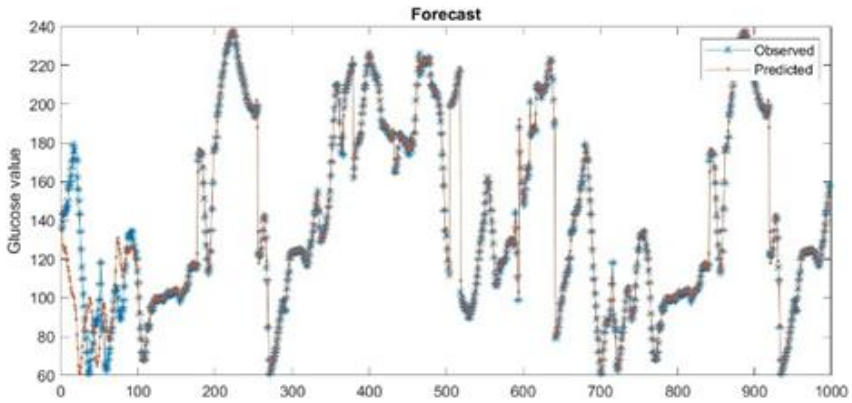


a)

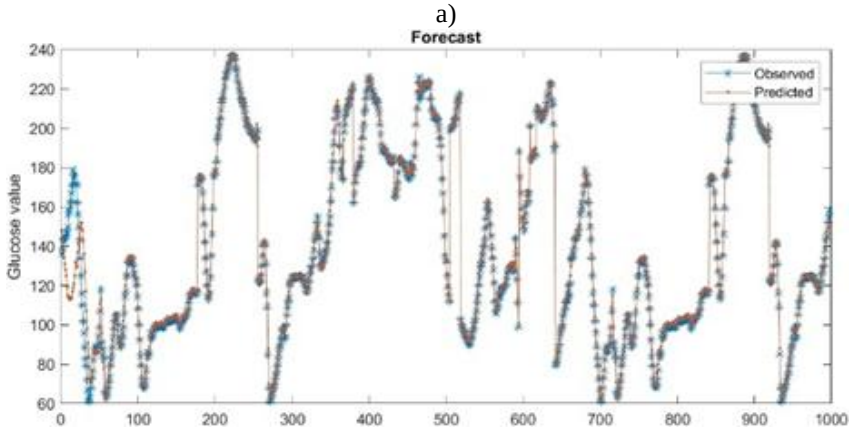


b)

Fig.3Blood glucose level predictions for 15 minutes PH of: a) single LSTM and b) stacked LSTM against the observed data

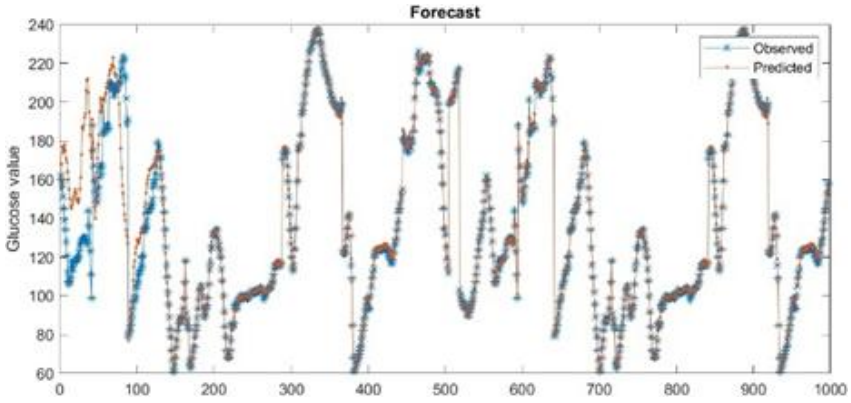


a)

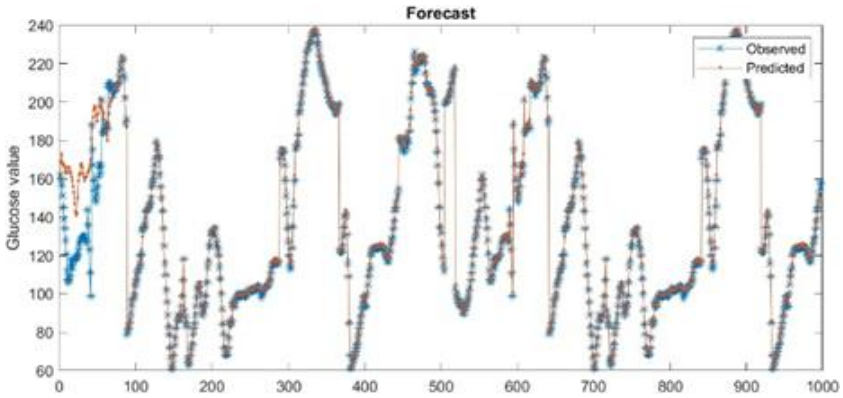


b)

Fig. 4Blood glucose level predictions for 30 minutes PH of: a) single LSTM and b) stacked LSTM against the observed data



a)



b)

Fig.5 Blood glucose level predictions for 45 minutes PH of: a) single LSTM and b) stacked LSTM against the observed data

V. CONCLUSION

In this paper, we explore the effectiveness and potential of using different LSTM recurrent neural networks for the blood glucose level prediction. In general, LSTM networks have shown promising results for time series predictions because of their ability to catch information from complex time series data.

Prediction accuracy was eventuated by RMSE and Pearson correlation coefficient. Obtained results showed better performance of stacked LSTM over single layer LSTM during each PH. However, increasing prediction horizon, the RMSE values for both the single and the stacked LSTM, increased. The best RMSEs are for 15 minutes prediction horizon.

Therefore, future work will be related to improve presented models for longer period of prediction as well as to develop and test other type of deep learning neural networks.

REFERENCES

[1] K. Ogurtsova,JD da Rocha Fernandes, Y. Huang, U. Linnenkamp, L. Guariguata, NH Cho, D. Cavan, JEShaw, andLEMakaroff,IDF Diabetes Atlas: Global estimates for the prevalence of diabetes for 2015 and 2040, (Diabetes Res Clin Pract, 2017), 40–50.

[2] M. M. Ahmadi and G. A. Jullien, A wireless-implantable microsystem for continuous blood glucose monitoring, *IEEE Transactions on Biomedical Circuits and Systems, vol. 3, no. 3*, (Jun. 2009), 169–180.

[3] J. Reifman, S. Rajaraman, A. Gribok, and W.K. Ward: Predictive monitoring for improved management of glucose levels, *J Diabet Sci Technol*, 1:478–86. doi: 10.1177/193229680700100405, (2007).

[4] 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 Trans Biomed Eng., 54:931–7. doi: 10.1109/TBME.2006.889774, (2007).

[5] J. Yang, L. Li, Y. Shi, X. Xie: An ARIMA Model with Adaptive Orders for Predicting Blood Glucose Concentrations and Hypoglycemia, IEEE J Biomed. Health Inform., (2018), 23(3):1251–60.

[6] E.I. Georga, V.C. Protopappas, D. Ardigo, M. Marina, I. Zavaroni, D. Polyzos, D.I. Fotiadis: Multivariate prediction of subcutaneous glucose concentration in type 1 diabetes patients based on support vector regression. IEEE journal of biomedical and health informatics, 17(1), (2013), 71-81.

[7] R. Bunescu N. Struble, C. Marling, J. Shubrook, F. Schwartz: Blood glucose level prediction using physiological models and support vector regression, In 12th International conference on machine learning and applications. IEEE, 1, (2013), 135–40.

[8] G. Mougiakakou, A. Prountzou, D. Iliopoulou, K. S. Nikita,A.Vazeou, C. S. Bartsocas: Neural network based glucose - insulin metabolism models for children with type1 diabetes, *Proceedings of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '06)*, (September 2006), pp. 3545–3548.

[9] S. M. Pappada, B. D. Cameron, and P. M. Rosman: Development of a neural network for prediction of glucose concentration in type 1 diabetes patients, *Journal of Diabetes Science and Technology, vol. 2, no. 5*, (2008), pp. 792–801.

[10] C. Perez-Gandıa, A. Facchinetti, G. Sparacino et al.: Artificial neural network algorithm for online glucose prediction from continuous glucose monitoring, *Diabetes Technology and Therapeutics, vol. 12, no. 1* ,(2010), pp. 81–88.

[11] T. Zhu, L. Kezhi, H. Pau, C. Jianwei, G. Pantelis: A Deep Learning Algorithm For Personalized Blood Glucose Prediction, Department of Electronic and Electrical Engineering, (Imperial College London, London SW5 7AZ, UK, 2018).

[12] A. Mohebbi, A.R. Johansen, N. Hansen, P.E. Christensen, J.M. Tarp, M.L. Jensen, H. Bengtsson, and M. Mørup: Short term blood glucose prediction based on continuous glucose monitoring data. arXiv preprint arXiv:2002.02805, (2020).

[13] J. Martinsson, A. Schliep, B. Eliasson, C. Meijner, and S. Persson, O. Mogren: Automatic blood glucose prediction with confidence using recurrent neural networks, *in: 3rd International Workshop on Knowledge Discovery in Healthcare Data*,(2018),pp. 64–68.

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

[15] K. Li, J. Daniels, C. Liu, P. Herrero-Vinas, P. Georgiou: Convolutional recurrent neural networks for glucose prediction. *IEEE Journal of Biomedical and Health Informatics*, (2019).

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

[17] J.Brownlee, Stacked Long Short-Term Memory Networks, (August 18, 2017).

[18] I.Grau, G.Napoles, I.Bonet, andM.Matilde Garcia, Backprogation through Time Algorithm for Training Reccurent Neural Networks using Variable Length Instances, (2012.)

[19] JaebCenter for Health Research, https://www.jaeb.org/projects/.

[20] T. Wang, W. Li, Blood glucose forecasting using LSTM variants under the context of open source artificial pancreas system, *in: 53rd Hawaii International Conference on System Sciences*, (Maui, Hawaii, 7-10 Jan 2020).

[21] F.Robby, Y.Tu, I.Hossen, I.Lee, A.S Maida, and X. Hei, Stacked LSTM Based Deep Reccurent Neural Network with Kalman Smoothing for Blood Glucose Prediction, *BMC Medical Informatics & Decision 21, 101*, (2021).