

# Blood Glucose Level Prediction as Time-Series Modeling using Sequence-to-Sequence Neural Networks

**Ananth Bhimireddy** <sup>1</sup> and **Priyanshu Sinha** <sup>2</sup> and **Bolu Oluwalade** <sup>3</sup> and **Judy Wawira Gichoya** <sup>4</sup>  
and **Saptarshi Purkayastha** <sup>5</sup>

**Abstract.** The management of blood glucose levels is critical in the care of Type 1 diabetes subjects. In extremes, high or low levels of blood glucose are fatal. To avoid such adverse events, there is the development and adoption of wearable technologies that continuously monitor blood glucose and administer insulin. This technology allows subjects to easily track their blood glucose levels with early intervention, preventing the need for hospital visits. The data collected from these sensors is an excellent candidate for the application of machine learning algorithms to learn patterns and predict future values of blood glucose levels. In this study, we developed artificial neural network algorithms based on the OhioT1DM training dataset that contains data on 12 subjects. The dataset contains features such as subject identifiers, continuous glucose monitoring data obtained in 5 minutes intervals, insulin infusion rate, etc. We developed individual models, including LSTM, BiLSTM, Convolutional LSTMs, TCN, and sequence-to-sequence models. We also developed transfer learning models based on the most important features of the data, as identified by a gradient boosting algorithm. These models were evaluated on the OhioT1DM test dataset that contains 6 unique subject's data. The model with the lowest RMSE values for the 30- and 60-minutes was selected as the best performing model. Our result shows that sequence-to-sequence BiLSTM performed better than the other models. This work demonstrates the potential of artificial neural networks algorithms in the management of Type 1 diabetes.

**Keywords.** Blood glucose prediction, Time-series model, Wearable devices, Transfer learning

## 1 INTRODUCTION

Diabetes Mellitus is a chronic disease characterized by high blood glucose levels. According to the 2020 CDC National Diabetes Statistics Report, about 34.2 million people or 10.2 percent of Americans have diabetes [3]. Diabetes is classified as Type 1, Type 2 and Gestational Diabetes. Insulin is an enzyme produced in the pancreas that helps in blood glucose absorption into cells. In Type 1 diabetes (T1DM), the pancreas produces little or no insulin. In contrast, in Type 2 diabetes (T2DM), the pancreas produces a small amount of

insulin, or the body is resistant to the effect of insulin. This absence or resistance to insulin leads to an increase in blood glucose levels, known as hyperglycemia. Symptoms of hyperglycemia include excessive thirst, excessive urination, sweating, etc. Diabetic ketoacidosis is a serious complication of uncontrolled hyperglycemia that can lead to death. On the other hand, elevated insulin levels in the body can cause low levels of blood glucose, a condition known as hypoglycemia. Dizziness, weakness, coma, or eventually, death can occur in uncontrolled hypoglycemia. Insulin and glucose control are critical to the management of diabetes, and hence titration of the administered insulin doses is critical in management of diabetes patients.

Glucose levels vary according to the patient's diet, and activities throughout the day. Sensors have been developed to estimate blood glucose levels at various time intervals. These sensors are useful in diabetes management because they provide longitudinal data about subjects' blood glucose and show the distinctive patterns throughout the day. These sensors are frequently coupled with the use of insulin pumps to deliver short-acting insulin continuously (basal rate) and specific insulin quantity after a meal for appropriate glycemic control. Although the sensors and insulin pumps have helped to improve patient care, patients are typically unaware of an impending adverse event of severe hyperglycemia or hypoglycemia. These adverse effects commonly occur when patients are asleep. There is an opportunity for the development of accurate prediction models using previously collected sensor data to estimate future values of blood glucose levels to prevent the occurrence of adverse events.

In this study, we utilize the OhioT1DM dataset, which contains blood glucose values of twelve T1DM subjects collected at intervals over a total time span of eight weeks [13]. These individuals had an insulin pump with continuous glucose monitoring (CGM), wearing a physical activity band and self-reporting life events using a smartphone application. CGM blood glucose data were obtained at 5-minute intervals [12]. We developed multiple models for predicting glucose values at 30 and 60 minutes in the future, using the CGM values, and mean Root Mean Square Error (RMSE) as an evaluation metric. The code for this study can be found at <https://github.com/iupui-soic/bglp2>

## 2 RELATED WORK

Models like LSTM and RNN have been used for forecasting in [10] which has been improved since then. The paper explores short-term load forecasting for individual electric customers and proposes an LSTM based framework to tackle the issue. Machine Learning models such as XGBoost have been used to predict glycemia in type-1 diabetic patients in [14]. This paper experiments primarily with

<sup>1</sup> Indiana University Purdue University Indianapolis, USA, email: anbhimi@iupui.edu

<sup>2</sup> Mentor Graphics India Pvt. Ltd., India, email: priyanshusinha@outlook.com

<sup>3</sup> Indiana University Purdue University Indianapolis, USA, email: boluwala@iupui.edu

<sup>4</sup> Emory University School of Medicine, USA, email: judywawira@emory.edu

<sup>5</sup> Indiana University Purdue University Indianapolis, USA, email: sapt-purk@iupui.edu

XGBoost algorithm to predict the blood glucose levels at a 30-minute horizon in the OhioT1DM dataset. Features from pre-trained TimeNets have been used for clinical predictions in [7]. This paper uses pre-training a network for some supervised or unsupervised tasks on datasets, and then fine-tuning via transfer learning for a related end-task to leverage the resources of labeled data in making predictions. This paper points out that training deep learning models such as RNNs and LSTMs requires large labeled data and is computationally expensive.

Deep learning models like Recurrent Neural Network (RNN) have been used on the OhioT1DM dataset to predict future blood glucose values [16], including the BGLP Challenge at KDH@IJCAI-ECAI 2018 (<http://ceur-ws.org/Vol-2148/>). In some cases, these data-driven models use only the CGM values or use physiological data such as the insulin concentration, amount of carbohydrate in meals and physical activities. Chen et al. created a data-driven 3-layer dilated recurrent neural network model with a mean RMSE of 18.9, with a range of 15.2995 and 22.7104 [4]. They concluded that the missing data and the continuous fluctuations in the data influenced the model's performance. Their model bettered the Convolutional Neural Network (CNN) that gave an average RMSE of 21.726 for six subjects [21]. Bertachi et al. predicted blood glucose levels using Artificial Neural Networks with the inclusion of physiological data [1]. Their results were not significantly dissimilar to those obtained in the data-driven models. All the previous studies demonstrate that a lower RMSE is obtained at the 30-minute prediction when compared to the 60-minutes values. We postulate that a hybrid approach of combining both the data-driven and physiological models could improve on the performance of the individual models and incorporate this in our approach.

## 3 METHODS

### 3.1 Dataset Description

A detailed description of the dataset has been previously published in the OhioT1DM dataset paper [12]. We used the data provided on 12 subjects for training, and 6 test subjects. Furthermore, the parameters basal and temp basal were merged into a single parameter.

We converted both the training and testing datasets from XML to CSV, preserving the time intervals. We did not use interpolation on the datasets as rules of the competition have prohibited interpolation. We tried using the forward and backward filling to fill the null values in the datasets, but they are creating additional time intervals which becomes a problem in testing datasets. So, no re-sampling technique was used in this paper to preserve the time intervals of the samples. The data pre-processing is performed on only 6 patients (test and train datasets) whose results are to be predicted. Subject-548 has the highest number of training records (12150) and Subject-552 has the lowest no. of records(9080). The no.of features varied from subject to subject which causes unevenness while training time series models. So, we have added features to subjects as required to ease the process of training and predicting. Missing data is handled in all columns by inputting the null values with zeros(0). We did not disturb the **glucose value** column as required by the competition.

### 3.2 Description of ML models

We used the following deep learning models to predict the blood glucose levels of each subject. The data pre-processing and model development are summarized in figure 1.

#### 3.2.1 Long Short-Term Memory (LSTM) Networks

LSTMs were originally introduced by Hochreiter and Schmidhuber [8] and later refined and popularized [17] [20]. LSTMs are a special kind of RNNs, capable of learning long-term dependencies. This quality of LSTMs helps memorize useful parts of the sequence and the model learns parameters more efficiently, making it useful for time series models.

We trained two models using LSTMs, one with 5-min interval data and another with 30-min interval data. In each model, we used all the available features at time  $t$  to predict the glucose value at time  $t+1$ . Before fitting the dataset, we scaled the dataset using MinMaxScaler from scikit-learn [15].

The LSTM model was built using the Keras [6] platform. We used 128 LSTM units, followed by a dense layer (150 units), dropout layer (0.20), dense layer (100 units), dropout layer (0.15), dense layer (50 units), dense layer (20 units) and a final layer with one unit (for prediction). We used *ReLU* as the activation function with *Adam* optimizer. The loss was calculated in *mean squared error (MSE)* and later converted into *Root Mean Squared Error (RMSE)*. The model was trained for 200 epochs with a batch size of 32. The results of the model for each subject are provided in the results table.

#### 3.2.2 Bi-directional Long Short-Term Memory (BiLSTM) Networks

As our input text data is static, and the entire sequence is available at the same time, we implemented a BiLSTM model to observe how processing the sequence from either direction affected the accuracy. The architecture of the BiLSTM model is similar to the LSTM model and is thus useful for time series prediction.

The data processing and model parameters for BiLSTM and LSTM model were similar with an exception in the model's first layer, where the scaled data was inputted into the Bidirectional LSTM with 128 units. The model was trained for 200 epochs with a batch size of 32. The results of the model for each subject are provided in the results table.

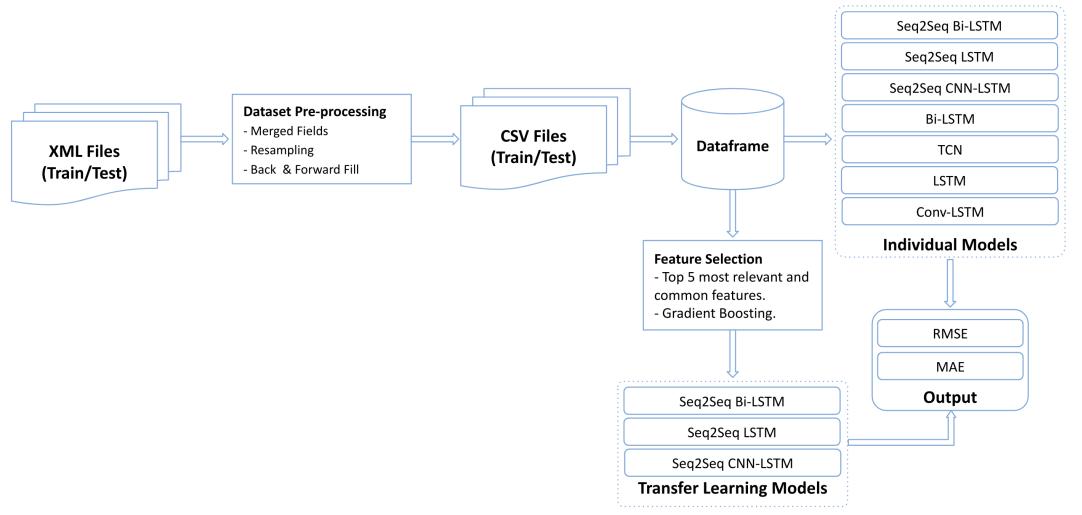
#### 3.2.3 Temporal Convolutional Networks (TCN)

TCNs were originally introduced by Lea and Videl [11] in 2016. TCNs are extremely useful in capturing the high-level temporal relationships in sequential networks. The TCN architecture allows capturing long-range spatio-temporal relationships. TCN's help capture the blood glucose level of subjects who usually have routine lifestyles, as TCNs can capture hierarchical relationships at low, intermediate, and high time scales.

The data processing steps are similar to the LSTM model. The TCN model was built using the Keras platform, but the depth of the model is relatively simpler compared to the LSTM and BiLSTM model. The scaled data was inputted into a TCN layer and then connected to a dense layer with one unit for the output. The model used *Adam* optimizer and *MSE* for calculating loss which was later converted to *RMSE*. The model was trained for 10 epochs and the obtained results are provided in the results table.

#### 3.2.4 Convolutional LSTM

Convolutional LSTMs (ConvLSTM) were introduced by Xingjian Shi et al. [18] in 2015. ConvLSTMs are created by extending the fully connected LSTM to have convolutional structure in both the input-to-state and state-to-state transitions. ConvLSTM network captures



**Figure 1.** Process Architecture

spatio-temporal correlations better and usually outperform Fully Connected LSTM networks.

The scaled data was reshaped and inputted to a convolution layer with 32 filters of kernel size 1, followed by LSTM layer with 128 units, Dense layer with 150 units, Dropout layer with 0.2 dropout rate, Dense layer with 100 units, Dropout layer with 0.15 rate, Dense layer with 50 units, Dense layer with 16 units and finally a Dense layer with 1 unit for prediction. *ReLU* activation function was used in all layers. The model used *Adam* optimizer and the *MSE* loss was further converted to RMSE for model comparison. The model was trained for 200 epochs with a batch size of 32. The obtained results are provided in the results table.

### 3.2.5 Description of Sequence-to-Sequence Models

Sequence to sequence models were first introduced by Google in 2014 [19]. These models map fixed-length input with fixed-length output where length of input and output differ. Sequence-to-sequence models consist of three parts:

- **Encoder:** Encoder consists of stacks of several recurrent (LSTM) units where each unit takes a single element of the input sequence, extracts information from it and propagates it to the next unit.
- **Encoded Vector:** This is the intermediate step and the final hidden layer of the encoder. It also acts as the first hidden layer for the decoder to make predictions. This vector encapsulates all information from all input samples and provides this information to the decoder.
- **Decoder:** This consists of stacks of recurrent unit where each unit predicts output at time step  $t$ . Each unit accepts a hidden layer as input and produces output as well as its own hidden state.

The main advantage of this architecture is that it can map sequences of different lengths to each other. We applied 3 variants of sequence to sequence models viz., sequence-to-sequence LSTM, sequence-to-sequence Bi-LSTM, and sequence-to-sequence CNN-LSTM.

For training of sequence-to-sequence models, we split the data into windows of 60 minutes. This approach is intuitive and helpful as BGLP values can be predicted 1-hour ahead. It is also helpful while modelling as the model can be used to predict blood glucose values at

specific time periods, let's say (after 10 minutes) or a whole sequence of blood glucose values.

To evaluate these sequence-to-sequence models we used walk forward validation. Here the model predicts the next one hour and then the actual data for one hour is given to make prediction for next one hour. See below table 1 for more illustration:

**Table 1.** Description of Sequence-to-Sequence input and prediction values

Input	Prediction
1st 60 minutes data	2nd 60 minutes data
[1st + 2nd] 60 minutes data	3rd 60 minutes data
[1st + 2nd + 3rd] 60 minutes data	4th 60 minutes data

For our training, we kept our input size (number of prior observations required to make next predictions) as 30 minutes data to predict next 60 minutes data. Each sequence-to-sequence model used in our work is described below:

#### 1. Sequence-to-Sequence LSTM

In this model, we used 200 LSTM cells for the encoder and decoder. This layer was followed with 2 dense layers containing 150 and 1 units wrapped in a *TimeDistributed* layer. The model was trained for 80 epochs with batch size of 40. We used *Adam* optimizer [9] with learning rate of 0.01 and loss function as *MSE*.

#### 2. Sequence-to-Sequence Bi-LSTM

In this model, we used 100 Bi-LSTM cells (LSTM cells wrapped in Bidirectional wrapper) for the encoder and decoder. This layer was followed with 2 dense layers containing 150 and 1 units wrapped in a *TimeDistributed* layer. The model was trained for 80 epochs with batch size of 40. Similar to sequence-to-sequence LSTM, Adam optimizer was used with 0.01 learning rate and “mean squared error” as the loss function.

#### 3. Sequence-to-Sequence CNN-LSTM

In this model, we used 2 1D Convolutional layer with filter size of 128 and 64 respectively. Convolutional layers were followed with MaxPooling 1D layer and flatten layer for the encoder. 200 LSTM cells are used for the decoder. This layer was followed with 2 dense layers containing 100 and 1 units wrapped in a *TimeDistributed* layer. The model was trained for 80 epochs with batch size of 40. Similar to

above models, Adam optimizer was used with 0.01 learning rate and “mean squared error” as the loss function.

### 3.2.6 Transfer Learning

For transfer learning, we first found the most relevant and common features among all 12 subjects. The importance of features was found using a **Gradient Boosting** algorithm. We set the cumulative frequency to 0.99 for feature selection and the 5 most important and common features are as follows: ‘finger stick value’, ‘basal rate value’, ‘galvanic skin response value’, ‘skin temperature value’, ‘bolus dose value’

Using the above-identified important features only, we trained our model on a randomly selected subject (567) and subsequently, fine-tuned each sequence-to-sequence model on each subject. The final model was used for prediction on the test data. The configurations of each model was similar to the sequence-to-sequence model as described above.

## 4 RESULTS

Figures 2 and 3 shows the comparison between the actual and the predicted values obtained from our best performing model i.e. sequence-to-sequence BiLSTM model. From the figures, it is clear that our model closely predicts the values of the test data by following similar peaks and troughs.

**Table 2.** RMSE and MAE results of Sequence-to-Sequence BiLSTM model

Subjects	RMSE		MAE	
	30 Mins	60 Mins	30 Mins	60 Mins
584	29.7	42.6	18.1	30.0
567	20.7	35.1	14.4	24.9
596	18.6	28.3	12.7	19.3
552	18.2	30.0	13.3	22.2
544	19.8	32.9	13.7	23.1
540	24.3	41.4	17.8	31.0
Mean	21.8	35.0	15.0	25.0
SD	4.0	5.4	2.1	4.2

The RMSE of the 30 minutes horizon predictions of the models are presented in tables 3, 4 and 5. From the tables, the sequence-to-sequence (Seq2Seq) models performed better than all the other models with an average RMSE of 23.5 for Seq-2-Seq LSTM, **21.8 for Seq-2-Seq BiLSTM**, and 23.0 for Seq-2-Seq CNN-LSTM. Table 2 describes the RMSE and MAE values for the sequence-to-sequence BiLSTM model which performed better than the individual and transfer learning models.

From Table 2, the value for the RMSE varies from 18.2 in subject 552 to 29.7 in subject 584 at 30 minutes, and between 28.3 in subject 596 to 42.6 in subject 584 at 60 minutes. The MAE values lies between 12.7 to 18.1 at 30 minutes and between 19.3 and 31.0 at 60 minutes.

Subject 584 in figure 3 shows more fluctuations, has a higher peak and lower troughs than subject 552 in figure 2. The effect of these variations are reflected in our results as patient 584 has the highest RMSE value while subject 552 has the lowest RMSE values. It is evident that the levels of variations in the individual subjects contributes significantly to the differences in RMSE values of the individual subjects in our model.

**Table 3.** RMSE values for individual models at 30 minutes Horizon

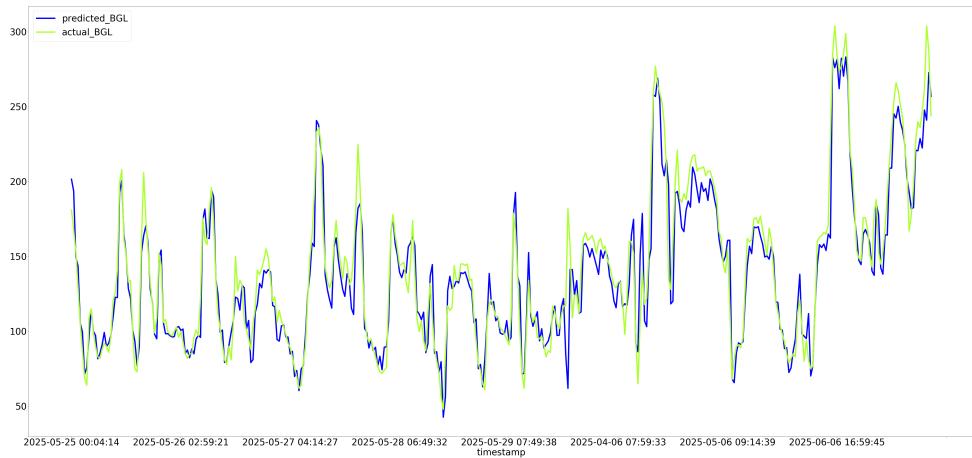
Subjects	LSTM	BiLSTM	TCN	ConvLSTM
584	27.97	29.06	26.55	26.57
567	25.65	27.04	25.71	26.04
596	19.47	20.30	18.95	21.08
552	20.62	20.30	17.14	17.73
544	21.34	22.06	80.67	20.94
540	30.21	31.72	25.94	27.35
Mean	25.0	24.4	34.7	23.2
SD	3.99	4.98	21.13	3.43

**Table 4.** RMSE values for sequence-to-sequence models at 30 Minutes horizon

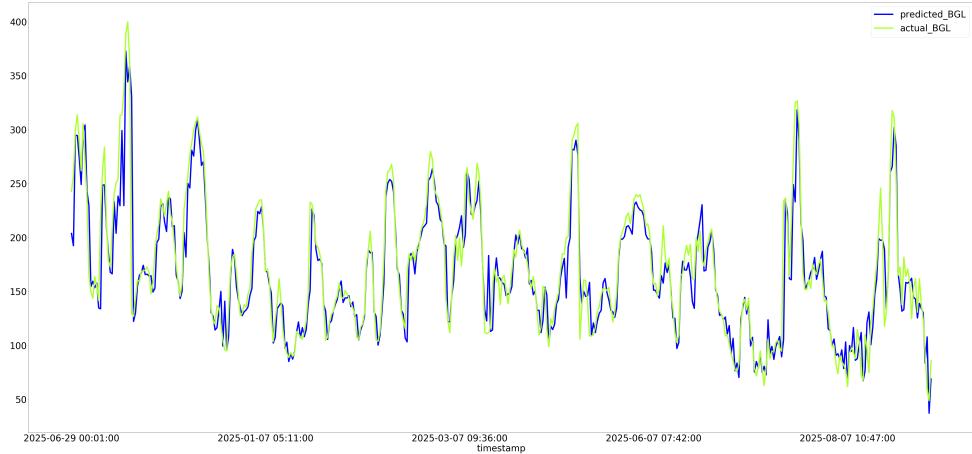
Subjects	Seq-2-Seq	Seq-2-Seq	Seq-2-Seq
	LSTM	BiLSTM	CNN-LSTM
567	29.2	20.7	29.0
540	25.0	24.3	23.1
544	18.5	19.8	19.2
596	18.6	18.6	19.0
584	30.7	29.7	31.0
552	19.2	18.1	17.2
Mean	23.5	21.8	23.0
SD	5.0	4.0	5.2

**Table 5.** RMSE values for sequence to Sequence transfer learning model at 30 Minutes horizon

Subjects	Seq-2-Seq	Seq-2-Seq	Seq-2-Seq
	LSTM	BiLSTM	CNN-LSTM
567	33.5	30.8	32.6
540	39.7	32.7	44.2
544	23.1	31.5	21.8
596	19.2	18.0	17.9
584	36.8	37.3	38.1
552	21.8	14.3	17.7
Mean	29.0	27.4	28.7
SD	7.9	8.3	10.2



**Figure 2.** Actual and Predicted values of subject 552 at 30 minutes horizons



**Figure 3.** Actual and Predicted values of subject 584 at 30 minutes horizons

## 5 CONCLUSION AND FUTURE SCOPE

In this paper, we present results of application of deep learning models to make predictions of blood glucose values. Potential benefits such as the prevention of adverse events associated extreme glucose values serve as a source of motivation for these efforts. Overall, sequence-to-sequence models especially Bi-LSTM have the best performance as these models are best at mapping sequences irrespective of their lengths. Our performance is affected by fluctuations in glucose values and also with missing data as described in previous experiments. Given the overall success of transfer learning, we also evaluated the potential of single model prediction via transfer learning approach. The transfer learning approach was inferior to the sequence to sequence models.

Compared to the previous paper for BGLP Challenge, we observed that two papers [2] and [5] have better results than our pro-

posed results. But [5] have used interpolation as a part of data processing which is against the rules of the competition and [2] did not mention details about data processing. Our future work will be to improve the transfer learning model as we are provided with more common features among all subjects, so that we can create a generic model for predicting blood glucose levels. However, the development of a generic model can be challenging because of confounding factors such as variations in sensor types, lifestyles, physiology and genetics. It is therefore pertinent that these factors are considered in future endeavors.

## REFERENCES

- [1] Arthur Bertachi, Lyvia Biagi, Iván Contreras, Ningsu Luo, and Josep Vehí, ‘Prediction of blood glucose levels and nocturnal hypoglycemia using physiological models and artificial neural networks.’, in *KHD@IJCAI*, pp. 85–90, (2018).

- [2] Arthur Bertachi, Lyvia Biagi, Iván Contreras, Ningsu Luo, and Josep Vehí, ‘Prediction of blood glucose levels and nocturnal hypoglycemia using physiological models and artificial neural networks’, in *KHD@IJCAI*, (2018).
- [3] CDC, *National Diabetes Statistics Report 2020. Estimates of diabetes and its burden in the United States.*, 2020.
- [4] Jianwei Chen, Kezhi Li, Pau Herrero, Taiyu Zhu, and Pantelis Georgiou, ‘Dilated recurrent neural network for short-time prediction of glucose concentration.’, in *KHD@ IJCAI*, pp. 69–73, (2018).
- [5] Jianwei Chen, Kezhi Li, Pau Herrero, Taiyu Zhu, and Pantelis Georgiou, ‘Dilated recurrent neural network for short-time prediction of glucose concentration’, in *KHD@IJCAI*, (2018).
- [6] François Chollet et al. Keras. <https://keras.io>, 2015.
- [7] Priyanka Gupta, Pankaj Malhotra, Lovekesh Vig, and Gautam Shroff, ‘Using features from pre-trained timenet for clinical predictions’, (07 2018).
- [8] Sepp Hochreiter and Jürgen Schmidhuber, ‘Long short-term memory’, *Neural Comput.*, **9**(8), 1735–1780, (November 1997).
- [9] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.
- [10] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, ‘Short-term residential load forecasting based on lstm recurrent neural network’, *IEEE Transactions on Smart Grid*, **10**(1), 841–851, (Jan 2019).
- [11] Colin Lea, René Vidal, Austin Reiter, and Gregory D. Hager, ‘Temporal convolutional networks: A unified approach to action segmentation’, *CoRR*, **abs/1608.08242**, (2016).
- [12] Cindy Marling and Razvan Bunescu, ‘The ohiot1dm dataset for blood glucose level prediction: Update 2020’.
- [13] Cindy Marling and Razvan C Bunescu, ‘The ohiot1dm dataset for blood glucose level prediction.’, in *KHD@ IJCAI*, pp. 60–63, (2018).
- [14] Cooper Midroni, Peter leimbigler, Gaurav baruah, maheeder kolla, alfred whitehead, and Yan Fossat, ‘Predicting glycemia in type 1 diabetes patients: Experiments with xgboost’, (07 2018).
- [15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, ‘Scikit-learn: Machine learning in Python’, *Journal of Machine Learning Research*, **12**, 2825–2830, (2011).
- [16] M Sangeetha and M Senthil Kumaran, ‘Deep learning-based data imputation on time-variant data using recurrent neural network’, *Soft Computing*, 1–12, (2020).
- [17] Alex Sherstinsky, ‘Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network’, *CoRR*, **abs/1808.03314**, (2018).
- [18] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo, ‘Convolutional LSTM network: A machine learning approach for precipitation nowcasting’, *CoRR*, **abs/1506.04214**, (2015).
- [19] Ilya Sutskever, Oriol Vinyals, and Quoc V Le, ‘Sequence to sequence learning with neural networks’, in *Advances in neural information processing systems*, pp. 3104–3112, (2014).
- [20] Peng Zhou, Zhenyu Qi, Suncong Zheng, Jiaming Xu, Hongyun Bao, and Bo Xu, ‘Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling’, in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 3485–3495, Osaka, Japan, (December 2016). The COLING 2016 Organizing Committee.
- [21] Taiyu Zhu, Kezhi Li, Pau Herrero, Jianwei Chen, and Pantelis Georgiou, ‘A deep learning algorithm for personalized blood glucose prediction.’, in *KHD@ IJCAI*, pp. 64–78, (2018).