Investigating potentials and pitfalls of knowledge distillation across datasets for blood glucose forecasting

Hadia Hameed, Samantha Kleinberg[1]

Abstract. Individuals with Type I diabetes (T1D) must frequently monitor their blood glucose (BG) and deliver insulin to regulate it. New devices like continuous glucose monitors (CGMs) and insulin pumps have helped reduce this burden by facilitating closed-loop technologies like the artifical pancreas (AP) for delivering insulin automatically. As more people use AP systems, which rely on a CGM and insulin pump, there has been a dramatic increase in the availability of large scale patient-generated health data (PGHD) in T1D. This data can potentially be used to train robust, generalizable models for accurate BG forecasting which can then be used to make forecasts for smaller datasets like OhioT1DM in real-time. In this work, we investigate the potential and pitfalls of using knowledge distillation to transfer knowledge from a model learned from one dataset to another and compare it with the baseline case of using either dataset alone. We show that using a pre-trained model to do BG forecasting for OhioT1DM from CGM data only (univariate setting) has comparable performance to training on OhioT1DM itself. Using a single-step, univariate recurrent neural network (RNN) trained on OhioT1DM data alone, we achieve an overall RMSE of 19.21 and 31.77 mg/dl for a prediction horizon (PH) of 30 and 60 minutes respectively.

1 Introduction

Type 1 diabetes (T1D) is a chronic lifelong disease that requires dozens of daily decisions to manage blood glucose (BG). While keeping BG in a healthy range is critical for avoiding complications, it is challenging, as meals and many other factors like exercise and stress can affect BG and insulin sensitivity. Closed-loop technologies, which connect a continuous glucose monitor (CGM) and insulin pump with a control algorithm, could relieve this burden by automatically dosing insulin. This requires an accurate forecast of where glucose is headed so the right amount of insulin can be delivered to keep BG within a target range dynamically.

Prior works include using system identification techniques to model glucose-insulin interactions [18, 3] , using classic autoregressive models for time series forecasting [23, 1, 5, 6] or training deep neural networks to implicitly learn the changing glucose level patterns [16, 17, 4, 24]. Neural network architectures such as LSTM have been used successfully for many time series forecasting problems [10, 8, 7, 19, 15], but require large amounts of training data. This is a challenge for BG forecasting, as it is time consuming and can be infeasible to collect such massive datasets. However, there are now large public datasets created by people with diabetes sharing their own data, which we believe could be leveraged. In particular, the open source artificial pancreas system (OAPS) [11], a collaborative project led by people with T1D, has data donated by individuals using the system. To date, there is open source diabetes data available for more than 100 subjects, collected over a period of 1 – 4 years (more than 1000 days worth of data for some individuals). This patient generated data is self-reported, noisy, heterogeneous, and irregularly sampled, but its much larger than the datasets routinely collected in controlled studies.

We propose that large public datasets like OAPS can be used to pretrain models, allowing deep learning to be used on smaller curated datasets for forecasing BG. In particular, we show by augmenting and distilling knowledge across models trained on data obtained from different sources using RNN, we achieve an accuracy comparable to that achieved by using OhioT1DM dataset alone for univariate setting. We also compare the performance with multi-output setting in which multiple BG values are estimated in the prediction horizon simultaneously. The code is available at https://github.com/health-ai-lab/BGLP_BG_forcasting.

2 Methodology

The task here is to forecast future values for BG. We compare single-step and multi-output forecasting. In the single-step setting, a single glucose value is estimated several minutes into the future, whereas in multi-output forecasting several future values are estimated simultaneously to model the signal trajectory over the prediction horizon. We begin by describing our time series forecasting approach, and later discuss the dataset specific preprocessing.

2.1 Problem setup

We define the feature vector $X_{0:t} = \{x_0, x_1, ..., x_t\} \in \mathbb{R}^n$ with n being the number of variables. We use only raw CGM values and do not incorporate additional features like carbohydrate intake and insulin dosage. We also have a corresponding output time series $X'_{t+1:t+h} = \{x'_{t+1}, x'_{t+2}, \ldots, x'_{t+h}\} \in \mathbb{R}$ representing multiple future glucose values across a given prediction horizon (PH) of 30 and 60 minutes. As CGM data is recorded at a frequency of 5 minutes, a PH of 30 and 60 minutes will lead to $h = 6$ and $h = 12$ samples, respectively. For the single step setting, this target vector becomes $X'_{t+h} = \{x'_{t+h}\}$ estimating only a single value h time instances in the future. Multi-output forecasting, on the other hand, aims to estimate the joint probability $p(X'_{t+1:t+h}|X_{0:t})$ simultaneously. However, root mean square error (RMSE) was calculated by comparing the actual future glucose level and the last future value in the estimated multi-output sequence, to accurately measure the performance of the forecasting model across the two output settings.

[1] Stevens Institute of Technology, USA, email: hhameed@stevens.edu

2.2 Learning Framework

Our proposed approach is to make glucose estimations for a small dataset by pre-training an RNN on a larger dataset and then re-training it using a smaller dataset. We compare four learning approaches for glucose forecasting, as shown in Fig.1: I) training and testing an RNN on OhioT1DM only (red path), II) training an RNN on OAPS dataset and testing on OhioT1DM without any re-training (blue path), III) training an RNN on OAPS dataset, training again OhioT1DM, and then testing on the OhioT1DM (purple path), and IV) the pre-trained RNN model makes intermediate estimates called soft predictions, which are given as target estimates to a student artificial neural network (ANN) model instead of the actual ground truth, as done for a classification task in [2]. As shown in the figure, the black edges from the two datasets to the teacher model show that it is pre-trained using the source data (OAPS here) but uses target data for making final predictions in Approach II, for re-training in Approach III, and making soft estimations in Approach IV (mimic learning), thus always having access to the two datasets.

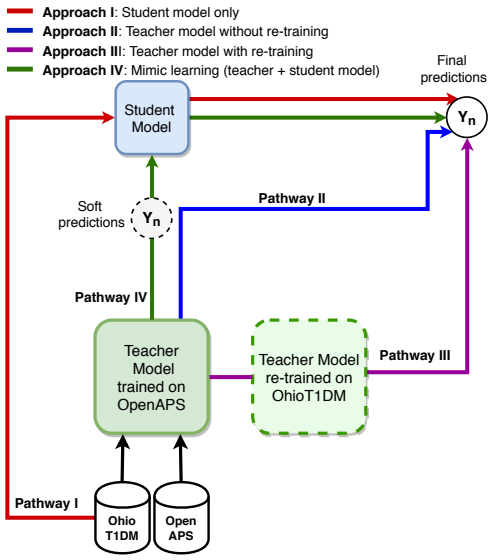

Figure 1: Four learning pipelines to estimate blood glucose levels in OhioT1DM test data.

2.3 Network Architecture

We use a vanilla RNN with a single hidden layer, $H(t)$ with 32 units, followed by a fully-connected output layer $O(t)$. This was used as the teacher model in Approaches II, III and IV and trained on the source patient-generated data. In Approach I where no teacher model was involved, the RNN was used as a student model trained only on the target OhioT1DM dataset to observe the effects of using datasets of different sizes and from different sources with the same network architecture. In Approach IV, a teacher RNN model trained on OAPS was used to teach a student ANN model using OhioT1DM data to study the effects of knowledge distillation between different kinds of networks. We use a simple, fully-connected ANN with a single hidden layer with 32 units. The number of units for both RNN and ANN were chosen after trying and testing [28, 32, 64, 128] and optimizing for the least RMSE. The output layer $O(t)$ predicts the glucose value(s) 30 or 60 minutes into the future depending on the PH and the output setting (single-step or multi-output).

2.4 Training models

The teacher model was trained on OAPS dataset which was pre-processed the same way as OhioT1DM, as discussed in Section 4. Early stopping was used to halt the training process if validation loss was not improving significantly, with the maximum number of epochs being 1000 with a batch size of 248 and 128 for OAPS and OhioT1DM, proportional to size of each dataset. Glorot normal initialization [9] was used to initialize the weight matrix. For the OhioT1DM dataset, the same training configurations (maximum epochs, batch size, initialization technique etc.) were used with all the learning approaches (i.e. student, teacher, retrained teacher, teacher-student) for a fair comparison. The experiments for Approach I, III and IV were repeated 10 times and the average RMSE and MAE was recorded for each subject, along with the standard deviation as presented in the Section 5.

3 Data

We aim to evaluate the impact of using a large noisy dataset for improving forecasting in a smaller more controlled dataset. The larger (source) dataset from OAPS [20] was used to pre-train the model before it was trained on OhioT1DM [12] (target), which is much smaller in terms of the total number of subjects and days for each.

3.1 OAPS

The collection of OAPS data started in 2015 as part of an initiative to make APS technology more accessible and transparent for people with T1D and to enable them to create their own customized AP systems. Participants can voluntarily donate their data, including glucose levels recorded via CGM, insulin basal and bolus rates, carbs intake, physical activity, and other physiological data. Researchers can gain access to this dataset free of charge, provided they share their insights and research findings with the public within a reasonable frame of time [21]. For this work, we used a subset of the dataset from individuals with multiple calendar years of data (55 people total, 320 ± 158.3 days of data on average). Since this data is largely self-reported, it is noisy, irregularly sampled, and heterogeneous in terms of the variables recorded, but because of its sheer size, it is highly useful for pre-training a robust machine learning model for accurate BG forecasting.

3.2 OhioT1DM

The training data consists of 12 subjects: six from the OhioT1DM dataset shared in 2018 for the First BGLP Challenge (Group I)[13], and six from the second BGLP Challenge 2020 (Group II)[12]. The validation and test samples are drawn from the last 10 days of data for subjects in Group I and Group II, respectively. The dataset contains around 8 weeks of data for 20 variables including raw CGM values, insulin basal and boluses, carbohydrate intake, exercise, and sleep.

4 Data pre-processing

For both OAPS and OhioT1DM, we use four recorded variables and one attribute derived from the raw glucose values. The list of features

used in the experiments includes raw CGM values (*glucose_level*, insulin basal rate (*basal* and *temp_basal*), bolus amount (*bolus*), carbs intake (*meal*), and difference between consecutive glucose values calculated during data pre-processing (*glucose_diff*). The first step in data pre-processing was to synchronize the multi-modality data by generating a single timestamp data field based on the timestamps for each of the four fields, generating an irregularly sampled multivariate time series.

In OAPS dataset, there were two types of gaps present in the data, first where both timestamp and glucose values were missing, and second where the timestamp was recorded but the corresponding glucose value was missing. In OhioT1DM, missing glucose values were identified once the multi-modality data was synchronized since basal, bolus, and meals are not recorded at the same 5-minute frequency as glucose levels. When there was missing glucose data for more than 25 consecutive minutes, these times were not used during training. Each data segment (series of points not separated by gap longer than 25 minutes) was then imputed and windowed separately to maintain temporal continuity in the data.

For the rest of the data, which may contain shorter gaps, we used linear interpolation to impute missing glucose values in training data. Missing values in test data were imputed by extrapolation to avoid using data from the future. Basal rates were imputed with forward filling, meaning replacing missing values with the last recorded basal rate, since the value is only recorded when it changes and thus missing values mean the last recorded one is still active. However, if the field "*temp_basal*", recording temporary basal infusion rate, was present for a given set of timestamps, it was used to replace the recorded basal rate [12] by evenly distributing the rate across the time duration which was divided into 5-minute intervals, as implemented in [14, 22]. Bolus rates were imputed in a similar manner by calculating the rate for every 5-minute interval and distributing it evenly across the specified duration, and was set to 0 when it was not recorded, thereby indicating that insulin was not bolused for those time instances. Similarly, the data field "*meal*" which recorded the amount of carbohydrate intake was set to 0 when it was missing.

In addition to missing data, the sensors are also noisy, leading to sudden changes in glucose levels, which can cause high variance in the learned model. To remove these spikes, the signal was passed through a median filter with a window size of 5 samples, as in [25]. This was only done for training data and not for the validation and test sets to test robustness of the model.

A sliding window was used to split the data into fixed sized sequences for further downstream analysis. There are three parameters for the moving window configuration: history window size (number of past samples to use for forecasting), prediction horizon (PH) and output window (how far into the future and how many future values to predict), and stride (number of samples to skip while sliding the window). An hour (12 samples) of past values were used to predict the glucose levels 30 and 60 minutes into the future (PH = 30, 60) with a unit stride, which means overlapping windows were used to partition the data.

In OhioT1DM train and test data, the raw CGM values range from 70 – 275 mg/dl and 75 – 290 mg/dl on average, respectively. To ensure that values of all the features were in the same range, insulin basal, bolus rates and carbs intake were normalized based on the minimum and maximum value of glucose levels using Min-Max Normalization.

5 Experiments

5.1 Experimental set up

The last ten days of data for subjects with ID 559, 563, 570, 575, 588 and 591 were used as validation set and test set was sampled from data for subjects 540, 544, 552, 567, 584, 596. The processing steps for the test data included linear extrapolation for imputing missing values and normalization. The test data was not passed through a median filter like the training set to see how robust the trained models were to unseen, noisy data. We use root mean square error (RMSE) and mean absolute error (MAE) to compare the predicted values with the actual ground truth to evaluate the model. MAE and RMSE can be expressed as,

$$MAE = \frac{1}{n}\sum_{n=1}^{n}|y_i - \hat{y_i}| \quad (1)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{n=1}^{n}(y_i - \hat{y_i})^2} \quad (2)$$

where y_i is true glucose level and $\hat{y_i}$ is estimated glucose level, both measured in mg/dl. We repeated the experiments 10 times and calculated the average RMSE and MAE for each subject across the ten trials. We also report the best, worst and mean RMSE (MAE) across all the subjects for each of the four pipelines using both single-step and multi-output models.

5.2 Results

Table 1: RMSE (MAE) for single-step forecasting with different learning pipelines for a PH of 30 minutes.

(a) Single-step

Subject ID	I	II	III	IV
540	19.55 (14.00)	20.32 (14.60)	20.36 (14.69)	20.46 (14.81)
544	16.56 (11.51)	17.84 (12.51)	17.50 (12.20)	17.92 (12.53)
552	15.04 (11.14)	16.17 (11.90)	15.72 (11.63)	16.20 (12.06)
567	23.07 (14.67)	24.09 (15.38)	23.91 (15.32)	24.74 (15.65)
584	25.19 (16.16)	26.47 (16.88)	26.97 (16.65)	26.83 (16.84)
596	15.85 (10.98)	17.24 (12.06)	16.50 (11.52)	17.50 (12.12)
Best	15.04 (11.14)	16.17 (11.90)	15.72 (11.63)	16.20 (12.06)
Worst	25.19 (16.16)	26.47 (16.88)	26.97 (16.65)	26.83 (16.84)
Average	**19.21 (13.07)**	20.36 (13.89)	20.16 (13.67)	20.61 (14.00)

(b) Multi-output

Subject ID	I	II	III	IV
540	20.30 (14.64)	20.41 (14.77)	20.36 (14.68)	20.55 (15.18)
544	17.61 (12.19)	18.07 (12.59)	17.68 (12.23)	18.41 (12.91)
552	15.68 (11.57)	15.98 (11.74)	15.66 (11.54)	16.06 (12.08)
567	23.94 (15.29)	24.88 (15.58)	23.66 (15.08)	24.47 (15.55)
584	26.61 (16.65)	26.29 (16.71)	25.82 (16.43)	26.70 (17.01)
596	16.46 (11.43)	17.17 (11.86)	16.54 (16.54)	17.57 (12.21)
Best	15.68 (11.57)	15.98 (11.74)	15.66 (11.54)	17.57 (12.21)
Worst	26.61 (16.65)	26.29 (16.71)	25.82 (16.43)	26.70 (17.01)
Average	20.10 (13.63)	20.46 (13.87)	19.95 (13.57)	20.63 (14.16)

I: Student model only, II: Teacher model without re-training,
III: Teacher model with re-training, IV: Mimic learning (teacher + student model)

The results for a PH of 30 and 60 minutes are shown in Tables 1 and 2, respectively.

Overall, approach I achieved the lowest RMSE (MAE) with 19.21 (13.07) for a PH of 30 minutes and 31.77 (23.09) for PH = 60 minutes. In this approach an RNN was trained only using the OhioT1DM data, using raw CGM values. The worst performance was from approach IV, where estimations made by a teacher model pre-trained on OpenAPS dataset were given as ground truth to student ANN model for training on OhioT1DM, as shown in Tables 1a and 2a. This approach did not improve the forecast accuracy as it did in [2]. It might be because [2] used this technique for a classification task of mortality prediction which involved predicting hard labels and evaluated performance using misclassification error instead of estimating continuous valued deviations from the ground truth as is the case in BG forecasting.

For BG forecasting using multi-output model, all approaches performed equally well, with approach I, II, and IV (student model, teacher and teacher_student model) giving the same RMSE on average. For approach II, the error did not worsen significantly, showing that pre-trained models can be used for making forecasts for OhioT1DM data in real-time, without having to set aside a portion of the dataset for retraining the model, an important consideration for smaller datasets. However, the RMSE improved slightly for approach II when the teacher model was retrained.

Approach IV

Table 2: RMSE (MAE) for single-step forecasting with different learning pipelines for a PH of 60 minutes.

(a) Single-step

Subject ID	I	II	III	IV
540	33.94 (25.40)	35.54 (26.74)	35.16 (26.94)	35.84 (27.35)
544	27.79 (20.34)	31.07 (22.45)	30.79 (22.84)	31.23 (22.69)
552	26.68 (20.15)	28.36 (21.08)	27.99 (21.33)	28.54 (21.53)
567	37.99 (26.50)	39.89 (27.76)	40.63 (28.47)	39.57 (28.09)
584	37.47 (27.00)	39.74 (27.87)	39.40 (27.60)	39.36 (27.85)
596	26.72 (19.12)	28.41 (20.42)	27.89 (20.31)	28.75 (20.83)
Best	26.68 (20.15)	28.36 (21.08)	27.89 (20.31)	28.54 (21.53)
Worst	37.99 (26.50)	39.89 (27.76)	40.63 (28.47)	39.57 (28.09)
Average	**31.77 (23.09)**	33.84 (24.39)	33.64 (24.58)	33.88 (24.72)

(b) Multi-output

Subject ID	I	II	III	IV
540	35.23 (26.94)	35.32 (26.99)	35.33 (27.06)	35.66 (27.05)
544	30.68 (22.83)	31.17 (22.74)	30.73 (22.79)	31.23 (22.84)
552	28.22 (21.57)	28.57 (21.56)	28.13 (21.43)	29.23 (21.98)
567	39.53 (28.03)	39.07 (27.95)	39.32 (21.43)	41.33 (28.57)
584	39.60(27.71)	39.43 (27.91)	39.43 (21.43)	39.07 (27.37)
596	27.92 (20.27)	28.48 (20.55)	28.13 (20.42)	28.81 (20.81)
Best	27.92 (20.27)	28.48 (20.55)	28.13 (20.42)	28.81 (20.81)
Worst	39.60 (27.71)	39.43 (27.91)	39.43 (21.43)	41.33 (28.57)
Average	33.53 (24.56)	33.67 (24.62)	33.516 (24.52)	34.22 (24.77)

I: Student model only, II: Teacher model without re-training,

III: Teacher model with re-training, IV: Mimic learning (teacher + student model)

6 Conclusion

In this work we have compared four different learning strategies for BG forecasting using two different datasets. We have shown that an RNN model pre-trained on a bigger dataset such as OpenAPS can be used directly to do BG forecasting for a smaller dataset like OhioT1DM when using CGM data only. We predicted BG levels 30 and 60 minutes into the future using single-step and multi-output models, using univariate BG data. Overall, a single-step RNN trained only on univariate data from OhioT1DM dataset achieved the least RMSE of 19.21 and 31.77 mg/dl for a PH of 30 and 60 minutes, respectively.

ACKNOWLEDGEMENTS

We would like to thank the referees for their comments, which helped improve this paper considerably. This work was supported in part by the NSF under award number 1915182, NIH under award number R01LM011826, and Fulbright Scholarship.

REFERENCES

[1] Ransford Henry Botwey, Elena Daskalaki, Peter Diem, and Stavroula G Mougiakakou, 'Multi-model data fusion to improve an early warning system for hypo-/hyperglycemic events', in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 4843–4846. IEEE, (2014).

[2] Zhengping Che, Sanjay Purushotham, Robinder Khemani, and Yan Liu, 'Interpretable deep models for icu outcome prediction', in *AMIA Annual Symposium Proceedings*, volume 2016, p. 371. American Medical Informatics Association, (2016).

[3] C Cobelli, G Nucci, and S Del Prato, 'A physiological simulation model of the glucose-insulin system', in *Proceedings of the First Joint BMES/EMBS Conference. 1999 IEEE Engineering in Medicine and Biology 21st Annual Conference and the 1999 Annual Fall Meeting of the Biomedical Engineering Society (Cat. N*, volume 2, pp. 999–vol. IEEE, (1999).

[4] J Fernandez de Canete, S Gonzalez-Perez, and JC Ramos-Diaz, 'Artificial neural networks for closed loop control of in silico and ad hoc type 1 diabetes', *Computer methods and programs in biomedicine*, **106**(1), 55–66, (2012).

[5] Meriyan Eren-Oruklu, Ali Cinar, Lauretta Quinn, and Donald Smith, 'Estimation of future glucose concentrations with subject-specific recursive linear models', *Diabetes technology & therapeutics*, **11**(4), 243–253, (2009).

[6] Adiwinata Gani, Andrei V Gribok, Yinghui Lu, W Kenneth Ward, Robert A Vigersky, and Jaques Reifman, 'Universal glucose models for predicting subcutaneous glucose concentration in humans', *IEEE Transactions on Information Technology in Biomedicine*, **14**(1), 157–165, (2009).

[7] André Gensler, Janosch Henze, Bernhard Sick, and Nils Raabe, 'Deep learning for solar power forecasting—an approach using autoencoder and lstm neural networks', in *2016 IEEE international conference on systems, man, and cybernetics (SMC)*, pp. 002858–002865. IEEE, (2016).

[8] Felix A Gers, Douglas Eck, and Jürgen Schmidhuber, 'Applying lstm to time series predictable through time-window approaches', in *Neural Nets WIRN Vietri-01*, 193–200, Springer, (2002).

[9] Xavier Glorot and Yoshua Bengio, 'Understanding the difficulty of training deep feedforward neural networks', in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256, (2010).

[10] Nikolay Laptev, Jason Yosinski, Li Erran Li, and Slawek Smyl, 'Time-series extreme event forecasting with neural networks at uber', in *International Conference on Machine Learning*, volume 34, pp. 1–5, (2017).

[11] Dana Lewis, Scott Leibrand, and OpenAPS Community, 'Real-world use of open source artificial pancreas systems', *Journal of diabetes science and technology*, **10**(6), 1411, (2016).

[12] Cindy Marling and Razvan Bunescu, 'The ohiot1dm dataset for blood glucose level prediction: Update 2020', in *KHD@ IJCAI*, (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 J Leimbigler, Gaurav Baruah, Maheedhar Kolla, Alfred J Whitehead, and Yan Fossat, 'Predicting glycemia in type 1 diabetes patients: experiments with xgboost', *heart*, **60**(90), 120, (2018).

[15] Sadegh Mirshekarian, Razvan Bunescu, Cindy Marling, and Frank Schwartz, 'Using lstms to learn physiological models of blood glucose behavior', in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 2887–2891. IEEE, (2017).

[16] Stavroula G Mougiakakou, Aikaterini Prountzou, Dimitra Iliopoulou, Konstantina S Nikita, Andriani Vazeou, and Christos S Bartsocas, 'Neural network based glucose-insulin metabolism models for children with type 1 diabetes', in *2006 International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 3545–3548. IEEE, (2006).

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

[18] Fredrik Ståhl and Rolf Johansson, 'Diabetes mellitus modeling and short-term prediction based on blood glucose measurements', *Mathematical biosciences*, **217**(2), 101–117, (2009).

[19] Qingnan Sun, Marko V Jankovic, Lia Bally, and Stavroula 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)*, pp. 1–5. IEEE, (2018).

[20] Open Artificial Pancreas System. Openaps. `https://openaps.org/what-is-openaps/`, 2015. [Online; accessed 10-Dec-2019].

[21] Open Artificial Pancreas System. Openaps research application. `https://tinyurl.com/oaps-application`, 2015. [Online; accessed 10-Dec-2019].

[22] Jinyu Xie and Qian Wang, 'Benchmark machine learning approaches with classical time series approaches on the blood glucose level prediction challenge.', in *KHD@ IJCAI*, pp. 97–102, (2018).

[23] Jun Yang, Lei Li, Yimeng Shi, and Xiaolei Xie, 'An arima model with adaptive orders for predicting blood glucose concentrations and hypoglycemia', *IEEE journal of biomedical and health informatics*, **23**(3), 1251–1260, (2018).

[24] Konstantia Zarkogianni, Konstantinos Mitsis, Eleni Litsa, M-T Arredondo, G Fico, Alessio Fioravanti, and Konstantina S Nikita, 'Comparative assessment of glucose prediction models for patients with type 1 diabetes mellitus applying sensors for glucose and physical activity monitoring', *Medical & biological engineering & computing*, **53**(12), 1333–1343, (2015).

[25] 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).