

Dilated Recurrent Neural Network for Short-Time Prediction of Glucose Concentration

Jianwei Chen*, Kezhi Li*, Pau Herrero, Taiyu Zhu, Pantelis Georgiou

Department of Electronic and Electrical Engineering, Imperial College London, London SW5 7AZ, UK
jianwei.chen17@imperial.ac.uk, kezhi.li@imperial.ac.uk * †

Abstract

Diabetes is one of the diseases affecting 415 million people in the world. Developing a robust blood glucose (BG) prediction model has a profound influence especially important for the diabetes management. Subjects with diabetes need to adjust insulin doses according to the blood glucose levels to maintain blood glucose in a target range. An accurate glucose level prediction is able to provide subjects with diabetes with the future glucose levels, so that proper actions could be taken to avoid short-term dangerous consequences or long-term complications. With the developing of continuous glucose monitoring (CGM) systems, the accuracy of predicting the glucose levels can be improved using the machine learning techniques. In this paper, a new deep learning technique, which is based on the Dilated Recurrent Neural Network (DRNN) model, is proposed to predict the future glucose levels for prediction horizon (PH) of 30 minutes. And the method also can be implemented in real-time prediction as well. The result reveals that using the dilated connection in the RNN network, it can improve the accuracy of short-time glucose predictions significantly (RMSE = 19.04 in the blood glucose level prediction (BGLP) on and only on all data points provided).

1 Introduction

The prediction of BG levels has always been a challenge because of the difficulty of modeling its nonlinearity and considering the effect of different life events. Machine learning (ML) reveals a new approach to modeling the BG levels compared with traditional approaches, such as AR [Sparacino *et al.*, 2007] and ARMR model [Sparacino *et al.*, ;

*This work is submitted to the Blood Glucose Level Prediction Challenge, the 27th International Joint Conference on Artificial Intelligence and the 23rd European Conference on Artificial Intelligence (IJCAI-ECAI 2018), International Workshop on Knowledge Discovery in Healthcare Data.

†This work is supported by EPSRC, the ARISES project. J. Chen and K. Li are the main contributors to the paper.

Eren-Oruklu *et al.*, 2009], and their extrapolation algorithmic derivatives [Eren-Oruklu *et al.*, 2009; Gani *et al.*, 2009], and methods regarding neural networks [Zecchin *et al.*, 2012].

In particular, the blood glucose level prediction (BGLP) challenge provides a platform for artificial intelligence (AI) researches to evaluate the performance of different types of ML approaches on the real data. The OhioT1DM dataset provided by BGLP challenge records the eight weeks CGMs data as well as the corresponding daily events from six type-1 diabetes patients, which are referred by ID 559, 563, 570, 575, 588 and 591 [Marling and Bunesco, 2018]. The CGM data has a sampling rate of every 5 minutes. Since the BGLP can be regarded as time series prediction problem, the natural structure of recurrent neural networks (RNN) provides remarkable performance on the prediction of BG levels [Alanis *et al.*, 2011]. Moreover, the BG levels will be affected by different daily events, such as insulin injected, meals and exercises. Different types of events may have different temporal effects on the change of BG levels. Therefore, the solution is inspired by the recent research by [Shiyu Chang and Huang, 2017], which reveals the Dilated RNN (DRNN) with multi-resolution dilated recurrent skip connections allows the networks to learn different temporal dependencies at different layers. Lastly, before feeding the data into the model, the interpolation, extrapolation and filtering techniques are utilized to process the data in order to fill the missing data in training and testing set and remove the potential noise. Please note that in the testing dataset, the future glucose data points are not used in the extrapolation. Thus the algorithm can also be useful in real-time applications.

2 Data Processing

2.1 Preprocessing

Firstly, according to [Zecchin *et al.*, 2012], the accuracy of prediction based on neural networks can be improved by exploiting information on meals. Thus, each input batch of the DRNN model consists of the past 1 hour (12 points of data) CGMs, insulin doses, carbohydrate intake and time index. The CGMs, insulin and carbohydrate intake are corresponding to the fields 'glucose_level', 'bolus' and 'bwz_carb_input' respectively in the OhioT1DM dataset. The time index represents the position of each CGM data in a day. Other fields in the dataset have also been tried in the experiment, such as ex-

ercise, heart rate and skin temperature, which do not have the significant effect on the accuracy of the model, but increasing the variance of the model. It is worthy to note that, for some insulin and carbohydrate intake information, the timestamps can not be exactly matched to the timestamps in CGM data. They are set to associate to the timestamps in the CGM data with the smallest time difference.

Secondly, the output of the model is the difference between the next 6th point and the 12th point of CGM data in the input batch, which corresponds to the BG changes for the PH = 30.

Lastly, in order to improve the performance of DRNN model, the first-order linear interpolation and first-order extrapolation are applied to the training and testing set, respectively. The median filter is used only in the training set. These techniques will be explained in the following sections in details. The data of subjects 591 and 575 have the considerable amount of missing CGM data, the combination of training data from all patients with different proportions is used in the training process. The idea comes from the transfer learning technique in machine learning. The results obtained during the experiment shows that it improves the model performance.

2.2 Interpolation and Extrapolation

There are lots of missing CGM data for different patients in both training and testing set. Without interpolation or extrapolation, the missing data will cause discontinuities in the CGM curve. Moreover, the fake peaks caused by discontinuities will highly degrade the performance of the model. Thus, the first-order linear interpolation and first-order extrapolation algorithm are applied to the training set and testing set in this project, respectively. Based on the experiment result, the performance of the first-order interpolation and first-order extrapolation are similar for the testing set. The extrapolation technique does not use the information of future values to fill the missing data. Thus, the testing set uses extrapolation technique to fill the missing data, which enables the real-time prediction. Different interpolation algorithms have been tested, such as cubic interpolation, but the first-order linear interpolation provides the best result for the given data.

Figure 1 shows an example of the linear interpolation. The zero values in the original CGM data represents the missing data. However, the missing data is discarded if the missed time interval is significantly large. The purpose of this step is to prevent the model from attempting to learn the interpolation part instead of the real trend of CGM data. The data before a long time missing interval will be connected to the next nearest value in order to prevent any fake peaks. Furthermore, the insulin and carbohydrate intake in the missing CGM interval is set to zero.

2.3 Median Filtering

The median filter is employed only for the training set after the interpolation process, in order to remove part of fast variations and some small spikes in the linear region of the curve, which might be the noise in the data. Moreover, the curve will become more smooth and the trend in CGM data will become more obvious as shown in Figure 2. However, the length of filter window needs to be carefully set, otherwise the data will

be ruined and the model can not properly learn from the data. The window size is set to be 5 after comparing the results for different window sizes.

2.4 Using Data from Multiple Subjects

For the subject 575, there are 72 missing gaps and many gaps are significantly large. The large gaps are discarded as discussed in the previous section, hence the training data of subject 575 are not long enough for the model to learn using ML techniques, which might also result in overfitting easily. Therefore, the mixture of several patient's training set is introduced, which increases the training set by combining the data from different patients with different contributions, and the generalization of the model can be improved. This idea comes from the transfer learning technique, which is popular in the deep learning techniques that makes use of other related dataset to train the objective neural network. In this work we use 50% of the target subject's data plus 10% of other subjects' data to train the model first, and then train the final model based on the whole training set of the target subject.

For example, for subject 575, 50% training data were used in the first phase. Different proportions of the training data from other subjects are used in the training process as well (normally we use 10% of training data from other subjects). For the second phase, all training data for subject 575 are used to train the final model. By using the transfer learning technique, the RMSE of subject 575 are decreased further by about 0.7 compared with the result from only using its own data. Moreover, it is found that this approach can also be applied to the subject 591 to improve the result.

2.5 Evaluation metric

The overall performance of the model is evaluated by the root-mean-square error (RMSE) between the prediction curve and original testing curve. Since the output of the model is the changes of BG after 30 minutes, the prediction curve should be the firstly constructed based on the model output. The RMSE is computed as (1),

$$RMSE = \sqrt{\frac{1}{N} \sum (\hat{y} - y)^2} \quad (1)$$

where \hat{y} is the predicted value, y is the original value and N is the total number of points. However, since the interpolation/extrapolation is applied to both training and testing data, the imputed values should be removed when evaluating the RMSE in the testing phase, which guarantees the prediction curve is compared with the original test data with the same length. The total testing points for each patient are summarized in Table 1.

3 DRNN Model

With the processed training and testing data, the rest of work is to design the structure of the DRNN model and to tune the hyperparameters to obtain the best results. In this section, the DRNN model will be briefly introduced. The training and testing phase will be explained. Lastly, the effect of different hyperparameters will be investigated. In terms of the software

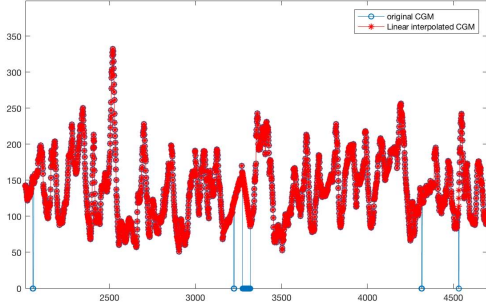


Figure 1: CGM data after linear interpolation.

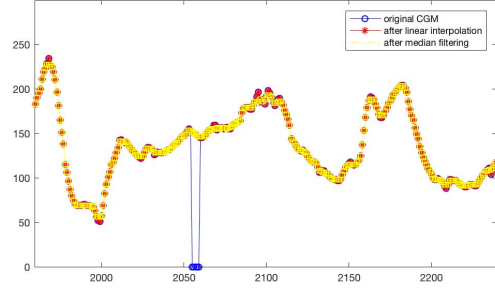


Figure 2: CGM data after linear interpolation and filtering.

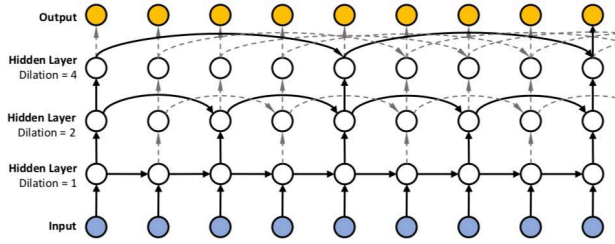


Figure 3: Structure of DRNN model. [Shiyu Chang et al., 2017]

implementation, the model is built based on tensorflow 1.6.0 and runs under the environment of python 3.6.4.

3.1 Model Structure

The DRNN model is characterised by its multi-resolution dilated recurrent skip connections as shown in Figure 3. The cell state for layer l at time t ($c_t^{(l)}$) is depending on the current input sample ($x_t^{(l)}$) and the cell state from $c_{t-s}^{(l)}$ as summarized in (2).

$$c_t^{(l)} = f\left(x_t^{(l)}, c_{t-s}^{(l)}\right), \quad (2)$$

where $x_t^{(l)}$ is the input to layer L at time t , s is the dilation and f represents the output function of different types of RNN cell, namely vanilla RNN, long short-term memory (LSTM) and gated recurrent unit (GRU). The multi-resolution dilated recurrent skip connections enable the model to capture the information for different temporal dependency and alleviate the vanishing gradient problem. The dilation is usually set to be increased exponentially [Shiyu Chang and Huang, 2017].

Patients	Total Testing Points
591	2760
570	2745
563	2570
559	2514
588	2791
575	2590

Table 1: Total testing points for each patient.

Therefore, the DRNN provides a powerful approach to process the long sequence data.

In this project, a 3-layered DRNN model is used, with 32 cells in each layer. Since the dilation is recommended to be the exponential increase, 1, 2 and 4 dilations are implemented for the 3 layers respectively from bottom to top.

3.2 Hyperparameters

Three different RNN cells have experimented, and the result shows that the vanilla RNN cell can achieve a better result than LSTM and GRU cells. Moreover, the training time and testing time using LSTM and GRU cells are significantly larger than vanilla RNN cell. This is because the structure of LSTM and GRU cells are much more complex than the vanilla RNN cell. Therefore, by implementing the vanilla RNN cell, better results can be obtained efficiently.

The effect of the number of cells in layers and the number of layers have also been investigated. It is found that the performance is degraded as the number of cells and layers increased. This is because the larger model requires relatively larger data set to converge. The training data points for each patient is around 10,000, which is not sufficient to train a large model properly. Therefore, a relatively small model as described is found to have a better performance.

3.3 Training and Testing

At each epoch of training step, the output from the model is used to reconstruct the prediction curve. Therefore, the RMSE is computed between prediction curve and the original curve. RMSProp optimizer [Ruder, 2017] with learning rate 0.001 is applied. Fixed batch size is set for all subjects. Varying the batch size also affects the accuracy of the model. In the experiment, it is found that larger batch size helps to improve the prediction results in term of RMSE.

When running the algorithm, an evaluation step is performed every 500 epoch. The advantage is that it provides a convenient way to monitor the training process and get the trend of accuracy, thus an appropriate number of epochs of training can be decided. Since the test data for each patient is around 2000 points, the cost of computation for the testing phase is relatively small. In this project, the 4000 to 5000 epochs is used in the training process. It should be noted that since the algorithm using past 12 points of data to predict the next 6th point (PH = 30), the last 17 points of the

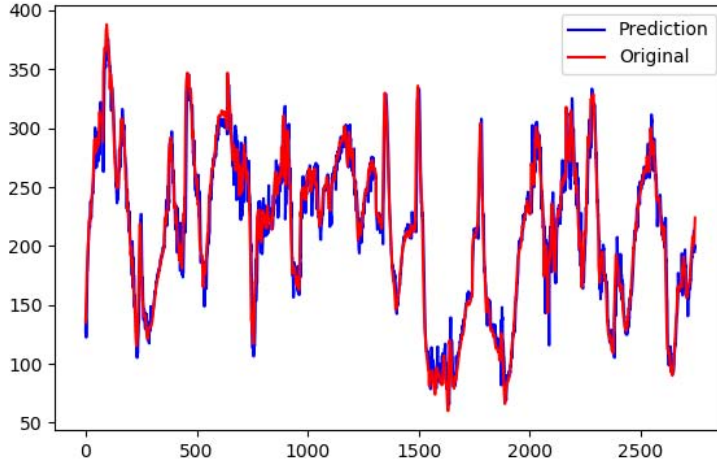


Figure 4: Predicted and Original test data of patient 570.

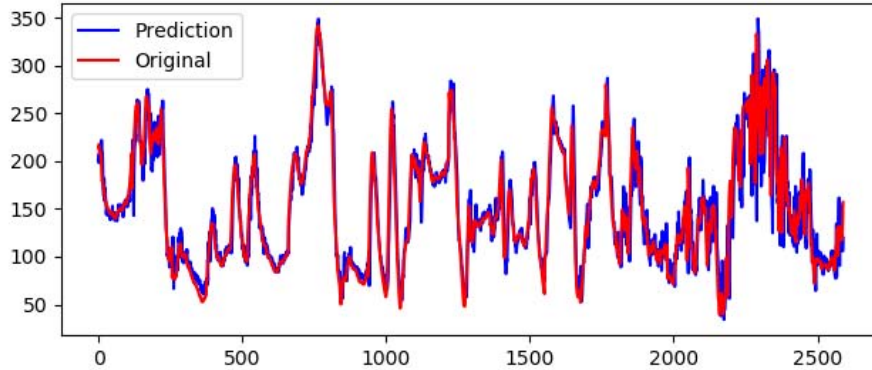


Figure 5: Predicted and Original test data of patient 575.

original training data should be appended at the beginning of the test set, which guarantees the length of prediction curve is the same as the original length of the test data (it has been approved by the BGLP challenge).

Patients	Best RMSE	Average RMSE \pm SD
591	21.1392	21.3354 \pm 0.0981
570	15.2995	15.4638 \pm 0.0907
563	18.0470	18.1153 \pm 0.0400
559	18.6158	18.7769 \pm 0.0672
588	17.6275	17.7270 \pm 0.0701
575	22.7104	22.8316 \pm 0.0941
Mean RMSE	18.9066	19.0417

Table 2: RMSE results of the DRNN model.

4 Experimental Results

With the data processed as described in Section 2 and model built as shown in Section 3, the RMSE of the test data is summarized in Table 2, where SD denotes the standard deviation. The best RMSE and average RMSE results are all based on 10 times simulation.

As can be seen from the Table 2, the RMSE for each subject vary from 22 to 15. The best RMSE is obtained in subject 570, and the RMSE is relatively large for subject 591 and 575. There are two reasons. Firstly, training data of 570 has relatively less missing data. There are 782 and 1309 missing data in subject 591 and 575, whereas subject 570 has 649 missing data. Secondly, through observing the curves of training and testing set for all subject, the data of subject 570 contains fewer fluctuations. The large fluctuation and continue peaks in both training and testing dataset will increase the difficulty of prediction, and degrade the model's learning capability, which can be observed from the result in Figure 5.

Figure 4 and Figure 5 show the prediction results of patient

570 and 575, which corresponds to the best RMSE shown in Table 2. As one can see that the test data of 575 is much fluctuant than 570, especially on the second half part of the curve.

More specifically, as shown in Figure 6, the relative linear region of the curve can be predicted with the small error. However, the fast and continues variations in the curve are almost impossible to predict, which contributes to a significant proportion of errors in terms of RMSE. Furthermore, a slight time delay in the prediction curve is observed, which is also a primary contribution of the errors.

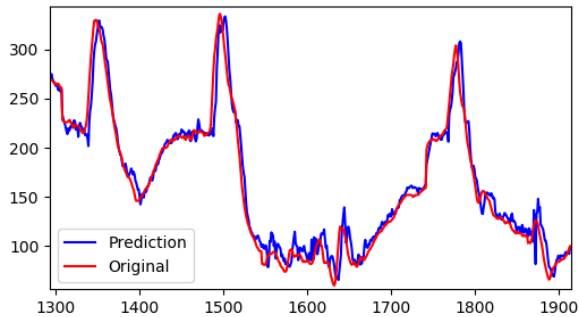


Figure 6: An example of prediction result.

5 Conclusion

This project aims to design an accurate short-time BG prediction model with $PH = 30$ minutes. The recent technique DRNN model has been exploited and applied in the project. The multi-resolution dilated recurrent skip connections of DRNN enables the network to learn different temporal dependencies in the sequential data. The data processing techniques of first-order linear interpolation, median filter, and a mixture of training set have been investigated. The results have shown the effectiveness of these data process methods.

With the DRNN model and data processing techniques, the performance of the whole algorithm is evaluated based on the OhioT1DM dataset. The RMSE results vary from 15.299 to 22.710 for different subjects with diabetes. More specifically, the missing data in the training and testing set, together with the fast continuous fluctuations in the data are the two main factors which degrade the accuracy of the model. In terms of improvement, there is still a large number of unused fields in the dataset. How to use these data properly and to feed them into the model still remain a challenge.

References

- [Alanis *et al.*, 2011] A. Y. Alanis, E. N. Sanchez, E. Ruiz-Velazquez, and B. S. Leon. Neural model of blood glucose level for type 1 diabetes mellitus patients. In *The 2011 International Joint Conference on Neural Networks*, pages 2018–2023, July 2011.
- [Eren-Oruklu *et al.*, 2009] Meriyan Eren-Oruklu, M.E., Ali Cinar, Lauretta Quinn, and Donald Smith. Estimation of future glucose concentrations with subject-specific recursive linear models. *Diabetes Technology and Therapeutics*, 11(4):243253, Apr. 2009.
- [Gani *et al.*, 2009] A. Gani, A. V. Gribok, S. Rajaraman, W. K. Ward, and J. Reifman *. Predicting subcutaneous glucose concentration in humans: Data-driven glucose modeling. *IEEE Transactions on Biomedical Engineering*, 56(2):246–254, Feb. 2009.
- [Marling and Bunescu, 2018] Cindy Marling and Razvan Bunescu. The ohioT1dm dataset for blood glucose level prediction. 2018.
- [Ruder, 2017] S. Ruder. An overview of gradient descent optimization algorithms. In *arXiv:1609.04747v2*. 2017.
- [Shiyu Chang and Huang, 2017] Wei Han Mo Yu Xiaoxiao Guo Wei Tan Xiaodong Cui Michael Witbrock Mark Hasegawa-Johnson Shiyu Chang, Yang Zhang and Thomas S. Huang. Dilated recurrent neural networks. In *31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA. 2017.
- [Sparacino *et al.*,] Giovanni Sparacino, Andrea Facchinetti, Alberto Maran, and Claudio Cobelli. Continuous glucose monitoring time series and hypo/hyperglycemia prevention: Requirements, methods, open problems. *Current Diabetes Reviews*, 4(3):181.
- [Sparacino *et al.*, 2007] G. Sparacino, F. Zanderigo, S. Corazza, A. Maran, A. Facchinetti, and C. Cobelli. Glucose concentration can be predicted ahead in time from continuous glucose monitoring sensor time-series. *IEEE Transactions on Biomedical Engineering*, 54(5):931–937, May 2007.
- [Zecchin *et al.*, 2012] C. Zecchin, A. Facchinetti, G. Sparacino, G. De Nicolao, and C. Cobelli. Neural network incorporating meal information improves accuracy of short-time prediction of glucose concentration. *IEEE Transactions on Biomedical Engineering*, 59(6):1550–1560, Jun. 2012.