# Neural Physiological Model: A Simple Module for Blood Glucose Prediction

Kang Gu , Ruoqi Dang, and Temiloluwa Prioleau

*Abstract*— **Continuous glucose monitors (CGM) and insulin pumps are becoming increasingly important in diabetes management. Additionally, data streams from these devices enable the prospect of accurate blood glucose prediction to support patients in preventing adverse glycemic events. In this paper, we present Neural Physiological Encoder (NPE), a simple module that leverages decomposed convolutional filters to automatically generate effective features that can be used with a downstream neural network for blood glucose prediction. To our knowledge, this is the first work to investigate a decomposed architecture in the diabetes domain. Our experimental results show that the proposed NPE model can effectively capture temporal patterns and blood glucose associations with other daily activities. For predicting blood glucose 30-mins in advance, NPE+LSTM yields an average root mean square error (RMSE) of 9.18 mg/dL on an in-house diabetes dataset from 34 subjects. Additionally, it achieves state-of-the-art RMSE of 17.80 mg/dL on a publicly available diabetes dataset (OhioT1DM) from 6 subjects.**

## I. Introduction

Diabetes is a prevalent chronic condition characterized by impaired glucose metabolism which yields frequent high and low blood glucose levels that increase the risk of macro- and micro-vascular complications [1]. Proper management of diabetes requires meticulous balance of various factors that affect blood glucose including food intake, medication, and activity [2]–[4]. However, innovative wearable medical devices are changing the standard of diabetes care. Several studies in literature have shown that continuous glucose monitors (CGM) and insulin pumps are useful to achieve better management of diabetes [5]–[7]. Given that CGMs and insulin pumps often work in isolation, significant research is being committed to develop closed-loop systems (i.e. artificial pancreas) that integrate data from these devices and others with the goal of automatically regulating blood glucose within the healthy range [5], [8], [9]. Materializing this vision requires effective methods for predicting blood glucose to inform appropriate timing of insulin delivery.

In prior research, various machine learning algorithms have been proposed for the task of blood glucose prediction including regression-based methods such as autoregressive integrated moving average (ARIMA) and support vector regression (SVR) [10]–[12]. However, traditional regression models do not include trainable weights and are not well-suited to take advantage of temporal dependencies present in blood glucose data. Hence, more sophisticated models like recurrent neural network (RNN) have been used in recent literature [13]–[15]. It is believed that the inherent structure of RNN is more fitting to capture the sequential correlations that are present in time-series data including blood glucose which is affected by food intake and insulin use.

Majority of the work in literature has focused on using only previous blood glucose data for prediction. While a few studies have explored the prospect of incorporating other related data from insulin pumps such as carbohydrate input, and insulin dosage/timing [14], [15]. However, a primary challenge to address when incorporating insulin pump data is the difference in data density. Based on today's technology, CGMs record a blood glucose sample every 5 minutes (i.e. 288 samples/day) while insulin pump data is more sparse [16]. Physiological models have been developed to inform how to combine CGM and insulin pump data using mathematical equations and variables to represent meal absorption, insulin dynamics and glucose dynamics [17], [18]. The work by Gu et al. [19] introduced multi-time-series deep LSTM model and found that the use of physiological models in combination with a long short-term memory (LSTM) network improved blood glucose prediction.

To circumvent the need for hand-crafted features and person-specific details in the standard physiological model [17], we introduce an approach called Neural Physiological Encoder (NPE) which can be trained jointly with any downstream neural network. The proposed NPE relies on decomposed convolution, which models the complex dependencies within physiological/daily-event data in two separate steps. First, the datastream is split into individual fields, with each group of filters extracting temporal features from the corresponding field respectively. Then the output of different groups are aggregated by convolutional kernels of size 1 to capture cross-event correlations. We have shown that prediction with the proposed NPE method outperforms traditional convolutional neural networks (CNN) by a distinctive margin on two independent datasets (i.e. an in-house dataset from 34 subjects and a publicly available dataset from 6 subjects).

## II. Related Work

Several review papers [10], [20], [21] can be referenced for a comprehensive understanding of efforts toward blood glucose prediction. However, in this section we focus on convolutional neural networks and recurrent neural networks.

### A. Recurrent Neural Networks

Given that the intrinsic architecture of RNN is sequential, many recent studies have leveraged RNNs to understand temporal patterns in CGM data. In [13], dilated recurrent neural networks (DRNN) was adapted to learn different temporal

Kang Gu and Temiloluwa Prioleau are with the Department of Computer Science, Dartmouth College, Hanover, NH, 03755, USA (email: [kang.gu.gr, temiloluwa.o.prioleau]@dartmouth.edu)

Ruoqi Dang is with Xi'an Jiaotong University, Xi'an, China (email: dangruoqi1999@stu.xjtu.edu.cn)

978-1-7281-1990-8/20/$31.00 ©2020 IEEE

dependencies of CGM data with multi-resolution dilated recurrent skip connections. The authors achieved a mean RMSE of 18.90 mg/dL for blood glucose prediction 30 minutes in advance on the publicly available OhioT1DM dataset [22]. Similarly, Lopez *et al.* [15] introduced *Gluco30p* to forecast glucose concentration along a short-term prediction horizon of 30 minutes. Using LSTM, a prevailing variant of RNN, they showed the advantages of training such a model on a large dataset and an accuracy of $> 90\%$ for predicting adverse glycemic events [15]. Most papers in literature strictly depend on previous blood glucose samples for prediction. In addition, a few papers [14], [15] have shown that the addition of insulin data is not beneficial for the task of glucose prediction. For example, adding insulin as an additional input feature only improved the prediction performance by $1\%$ in Lopez et al.'s work [15].

An obvious challenge that may be limiting these efforts relates to the approach of combining insulin data which is very sparse with blood glucose data which is more dense. Physiological models [23], [24] have been used successfully with LSTM to convert sparse insulin data into a continuous model; this approach provided a decrease in glucose prediction RMSE from 16.38 mg/dL to 14.04 mg/dL [19]. However, a sophisticated physiological model requires subject-specific information (e.g. weight, insulin sensitivity) and professional input to determine variables. To circumvent this need, we propose a Neural Physiological Encoder (NPE), that can be plugged into any neural network and trained end-to-end. NPE is capable of encoding raw data into discriminate features for succeeding modules. We test and compare combinations of NPE with three common types of RNNs, namely vanilla RNN, Gated Recurrent Unit (GRU), and LSTM.

### B. Convolutional Neural Networks

Convolutional neural networks (CNN) has also been used in the task of blood glucose prediction. For example, a CNN model that employed fast WaveNet algorithms was used to classify future glucose values into one of 256 quantized levels/classes [25]. This yielded a mean RMSE of 21.72 mg/dL on the publicly available OhioT1DM dataset [22]. Furthermore, Li et al. [26] employed a convolutional recurrent neural network (CRNN), a hybrid of two types of neural networks, where CNN worked as a feature extractor and RNN captured temporal dynamics for blood glucose prediction. However, their model achieved a similar mean RMSE of 21.07 mg/dL.

Unlike the aforementioned methods, our proposed NPE model decomposes the encoding process of CNN into two parts. Given the multidimensional input, each field of signal is fed into the corresponding convolutional block, which can exploit unique temporal correlations. Next, cross-field convolutional kernels are used to extract informative features by assigning larger weights to discriminative fields. Several papers [27]–[29] on decomposed convolution have shown that this improved structure is useful to improve the accuracy and efficiency of CNN. In this work, we employ decomposed convolutions to elucidate underlying correlations between daily events and physiological data by explicitly specifying temporal information learning filters and inter-activity dependencies learning filters.

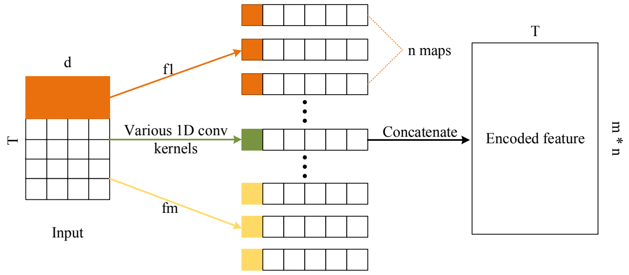

Fig. 1. The illustration of Baseline CNN. The input is a $d$ dimensional feature of $T$ time steps. A kernel library of $m$ sizes is applied to encode the input. The output representation will be $T \times (m * n)$ dimensional as each kernel instance will be randomly initialized $n$ times.

## III. Methodology

In this section, we present a baseline CNN module, then introduce the design of NPE (decomposed convolution) employed in this work.

### A. Baseline CNN

This module is composed of multi-channel 1D convolution blocks as shown in Fig. 1. The idea of expanding the width of a network was introduced in [30]. It was shown that applying convolutional kernels of varying sizes can greatly enrich the output by stacking multi-scale features produced by those kernels independently. In the task of blood glucose prediction, the benefit of employing multi-size filters is twofold. First, kernels of different window sizes are capable of encoding patterns of different time scales respectively; this can improve the performance of a subsequent network (e.g. LSTM). Secondly, as described in section II, kernels work as smoothing filters for the sparse fields of input data (i.e. insulin pump data).

Given $X_{1:T}$ as an input temporal sequence, we apply a group of 1D convolution kernels $[f_1, f_2, ..., f_m]$ to process this input using a constant number $n$ filters. Padding operation is performed so that the input length doesn't change, then the output vector of each convolution kernel is stacked to form a final encoded feature $\hat{X}$, $\hat{X} \in R^{T \times (m*n)}$.

### B. Neural Physiological Encoder

The baseline CNN encodes temporal and inter-field information in one step, but its performance is compromised by learning to extract the two kinds of information simultaneously. Fig. 2 shows the proposed hierarchical architecture of NPE which is designed to specify the function of convolutional kernels explicitly. This architecture uses a baseline CNN block to extract temporal information in each field, following that inter-field kernels are used to unwind the dependencies among the input fields. With this approach, the filters of each part is only optimized for a unique purpose. This can lead to improved accuracy and efficiency.

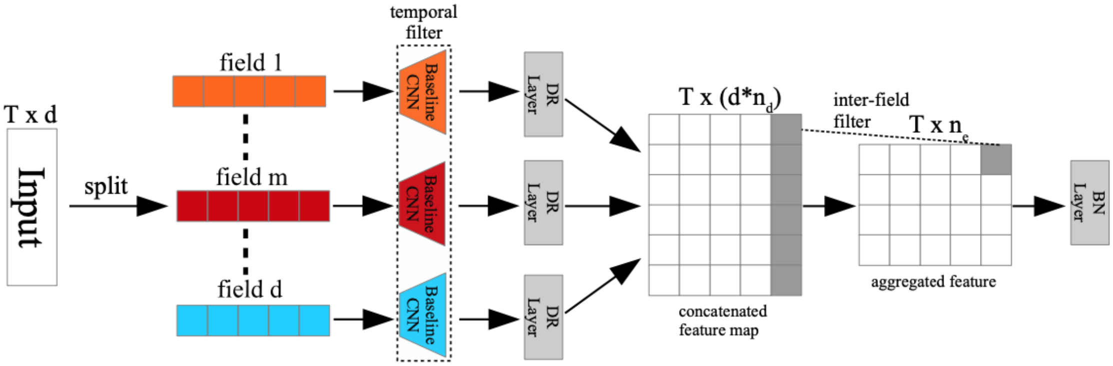

Fig. 2. An illustration of Neural Physiological Encoder. The input is a $d$ dimensional feature of $T$ time steps. A group of $d$ baseline CNN blocks are tailored for $d$ fields of input signal respectively. Following this, dimension reduction (DR) is achieved with 1D convolutional layers using $n_d$ kernels of size 1. Given the output dimension of each block is $n_d$, the concatenated representation will be $T \times (d * n_d)$ dimensions. Finally, $n_e$ cross-event filters are employed to aggregate the daily-event/physiological signals and this is fed into a batch normalization (BR) layer.

TABLE I
HYPERPARAMETERS FOR CNN & NPE

| Module | Parameter | Dataset-1 | Dataset-2 |
|---|---|---|---|
| CNN | $n$ (# of each filter) | 5 | 5 |
| NPE | $n_t$ (# of each temporal filter) | 1 | 1 |
| | $n_e$ (# of inter-field filter) | 10 | 20 |
| | $n_d$ (# of filter in DR layer) | 5 | 5 |
| Both | $w$ (filter's temporal window) | chosen from $[2, 3, 4, 5]$ | |

Similarly, we assume $X_{1:T}$ is the $d$ dimensional input temporal sequence. The output dimensionality of NPE is $k$, which is decided by the number of inter-field filters. Batch normalization (BN) layer is applied to normalize the aggregated feature map at the end [31], which accelerates the convergence of model.

### C. Computational cost of CNN and NPE

As shown in Table I, a fixed window size was chosen from [2,3,4,5] in both the baseline CNN and the proposed NPE model. A comparison of the number of parameters used in the baseline CNN and NPE model is as follows. For the baseline CNN, given $d$ features and $n$ filters, the total number of weights is $(14d + 4)n$ (i.e. approximately $14nd$). Meanwhile, for the NPE model, let $n_t$ be the number of temporal filters, $n_e$ be the number of cross-event filters, $n_d$ be the number of filters in DR layer. Then, the total number of weights, added up from the three parts is: $(14d + 4)dn_t + (4n_t n_d + 1)d + (dn_d n_e + 1) = (14d + 4n_d + 4)dn_t + d(n_d n_e + 1) + 1$. Since $n_t, n_d, n_e$ are usually fairly small, regardless of the growth of $d$, the number of parameters in NPE could be approximated as $14n_t d^2$. The ratio between the two estimations is $\frac{n}{n_t d} = \frac{n/n_t}{d}$, so if $d \leq n/n_t$, then NPE is actually a more efficient approach. As long as $d$ is in the range of 1 to 10, the NPE still has much less parameters than the downstream model (e.g. RNN).

## IV. NETWORK ARCHITECTURE

Fig 3 shows the computational flow of our proposed model. Input $X_{1:T}$ is composed of state $x_t$, where we set $T$ to be 108 for analysis on our in-house dataset and 12 for analysis on the OhioT1DM dataset. These values were chosen empirically from preliminary analysis on both datasets. Hence, $x_t = [g(t), e_1(t), e_2(t), ..., e_{d-1}(t)]$ is a $d$-dimensional vector representing the glucose value ($g(t)$) from a wearable CGM and events ($e_d(t)$) observed from other wearable devices at $t_{th}$ point. First, the multi-dimensional signal $X_{1:T}$ is fed into the head component, either baseline CNN or our proposed NPE architecture. The encoded feature after dimension reduction is $\hat{X}$, $\hat{X} \in R^{T \times k}$, where $k$ is set to be 5. Following this, we leverage a sequential network, such as RNN, gated recurrent unit (GRU) and LSTM which are evaluated in this work, to process the encoded features $\hat{X}$, generating an output of $h$ dimensions. Lastly, this is inputted into a fully connected layer to achieve a 1 dimensional scalar value. This model was trained to predict the glucose value at $T+1$ time step. However, for prediction, we append the output to the input until we reach the target prediction horizon (i.e. 30 minutes in this work).

### A. Training Objective

Similar to the prior work [13], [15], [25], [26], these models are trained to minimize the mean square error (MSE) between predicted value and ground-truth value:

$$e(\theta) = \frac{1}{N} \sum_{i=1}^{N} (y_{i,PH+j} - \hat{y}_{i,PH+j})^2 \qquad (1)$$

where PH is the prediction horizon, N stands for the number of samples in a batch, and $j$ is the start point. $\theta$ is the trainable parameters of the model.

## V. EXPERIMENT

In this section, we describe the datasets and preprocessing steps used in this work.

### A. Data Description

*1) Dataset-1:* This dataset was contributed to the *Digital SMD* project by members of an online open-source diabetes

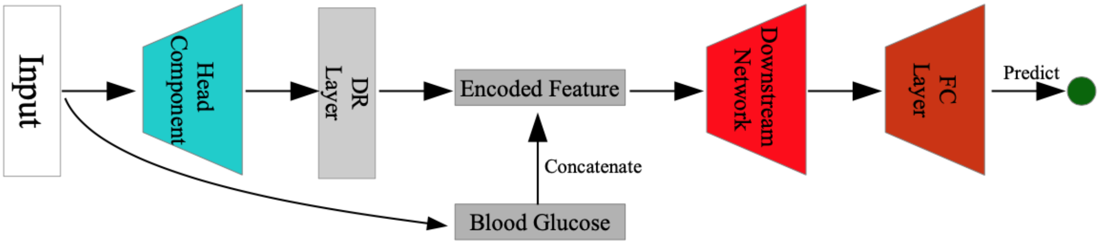

Fig. 3. The illustration of the overall architecture. The input is a $d$ dimensional feature of $T$ time steps. The head component encodes the temporal as well as inter-event dependencies, and DR layer compress the dimensionality to reduce computation cost. Then the encoded feature and blood glucose field are concatenated to form the input for downstream network (e.g. RNN). Finally, FC layer transforms the output of sequential network into blood glucose prediction.

TABLE II

HYPERPARAMETERS FOR TRAINING

| Parameter | Parameter Name | Dataset-1 | Dataset-2 |
|---|---|---|---|
| $d$ | feature numbers | 4 | 19 |
| $T$ | sequence length | 108 | 12 |
| $\lambda$ | coefficient of $L_2$-penalty | 0.01 | 0.01 |
| $\eta$ | learning rate | 0.001 | 0.001 |
| $Epoch$ | train epoch | 1000 | 300 |
| $Batch$ | batch size | 128 | 256 |
| $h$ | hidden units of RNNs | 300 | 128 |

community[1].This dataset includes 102 to 268 days of CGM and insulin pump data from 34 subjects with diabetes. Based on today's technology, CGM data is recorded at a frequency of $1/300$Hz (i.e. every 5 minutes). Additionally, the insulin pump data includes carbohydrate inputs from subjects associated with food intake, dosage amounts of basal and bolus insulin, and time stamps for each throughout the recording period. In this work, we used the full dataset from all 34 subjects to evaluate our model. For each subject, $80\%$ of the days was used for training and the remaining $20\%$ are reserved for testing. Our models were optimized on the training subset to search for proper hyperparameters, and evaluated on the test subset. Table II summarizes the selected hyperparameters used in this work.

*2) Dataset-2:* This refers to the publicly available OhioT1DM dataset [22] that consists of eight weeks of CGM, insulin, fitness tracker, and self-reported life-events data from 6 subjects with type 1 diabetes (ages: 40 to 60 yrs.). Self-reported life-event data (e.g. work) was collected via a smartphone app and physiological data (e.g. heart rate) was collected with a Basis fitness band. In total, OhioT1DM includes 19 features/data streams. This dataset was already split into a train and test set which was used in this work to facilitate fair comparison with related research papers.

## B. Data Preprocessing

For Dataset-1, we did not do any filtering or interpolation on the CGM data as the raw data was less noisy with limited ($<$ 30%) of missing data. With this dataset, the input to models described in this work consists of four dimensions, namely: 1) glucose_level, 2) basal_insulin_dose, 3) bolus_insulin_dose, and 4) carbohydrate_input. In rare cases when missing data occurred in the input data, we discarded the incomplete sequence.

Other the other hand, there are a lot of missing CGM values in both the training and test set of dataset-2. Informed by prior work [13], we performed a first-order interpolation and first-order extrapolation on the training and test sets, respectively. A median filter was then employed to remove false spikes after the interpolation process. With this dataset, the input to models described in this work consists of 19 dimensions, including glucose_level, basal, bolus, self-reported carbohydrate estimates, and other related life-event data which are described in the original paper [22]. Similarly, the selected hyperparameters associated with this dataset are shown in Table II.

## C. Other Implementation Details

*1) Prevent Overfitting:* Weight Decay and Dropout [32] were both used to prevent overfitting which is a common situation with smaller datasets. The weight decay coefficient ($L_2$ regularization) and dropout rate of neurons were set to 0.001 and 0.1, respectively.

*2) Optimization Method:* A method for stochastic optimization called Adam [33] was used with the learning rate set to 0.001 and exponential decay rate set to 0.9.

# VI. RESULTS AND DISCUSSIONS

Table III summarizes all related models employed in our experiments. Note that all the downstream networks share the same configuration. The hidden unit size is $d$, with recurrent weight decay of $0.01$ and dropout rate of $0.1$. Moreover, the parameters of CNN and NPE is presented in Table I. Each model was trained with all the features in a given dataset.

## A. Dataset-1

Table IV shows the results of various model combinations. Our proposed NPE+LSTM model outperformed other combinations by a distinctive margin. Specifically, NPE+LSTM achieves an RMSE of 9.18 mg/dL, reducing the RMSE

[1] http:// www.tidepool.org/

TABLE III

Competitor Models

| Head Component | Downstream Network | Full Model |
|---|---|---|
| None | RNN | RNN |
| | GRU | GRU |
| | LSTM | LSTM |
| Baseline CNN | RNN | CNN+RNN |
| | GRU | CNN+GRU |
| | LSTM | CNN+LSTM |
| NPE | RNN | NPE+RNN |
| | GRU | NPE+GRU |
| | LSTM | NPE+LSTM |

TABLE IV

The Prediction Error on Dataset-1

| | | *Average RMSE(mg/dL)* |
|---|---|---|
| RMSE | RNN | 18.36 |
| | GRU | 19.26 |
| | LSTM | 18.9 |
| | CNN+RNN | 16.02 |
| | CNN+GRU | 15.48 |
| | CNN+LSTM | 16.56 |
| | NPE+RNN | 13.32 |
| | NPE+GRU | 13.14 |
| | NPE+LSTM | 9.18$^\dagger$ |

by 3.96 mg/dL when compared to the second best model, NPE+GRU. It is important to notice that all combinations that includes our proposed NPE model performed better other single or combination models. This shows that NPE is fitting for feature encoding. On average, CNN improved the RMSE of the downstream network by $\approx$ 2.82 mg/dL, while NPE reduced the RMSE by $\approx$ 6.96 mg/dL. Even NPE+RNN, the worst in NPE group, exceeds CNN+GRU, the best in CNN group, by 2.16 mg/dL on RMSE metric. This shows that the decomposed convolution architecture in NPE can capture both temporal and cross-event dependencies in the dataset. Meanwhile, the baseline CNN's performance is constrained by the multitasking filter which cannot maximally extract temporal and cross-event correlations.

### B. Dataset-2

Table V summarizes the results obtained on this dataset; this results are shown per subject to further facilitate comparison with related works. However, the results are similar such that on average our proposed NPE+LSTM achieves the best and lowest RMSE of 17.80 mg/dL. This outperforms the best candidate in the CNN group by 2.12 mg/dL. Moreover, our NPE component is more robust and efficient since it boosts the performance of downstream network remarkably in almost every case, while CNN fails to enhance subsequent GRU or LSTM models. Even for CNN+RNN, the only effective combination in the CNN group, the RMSE only drops by 1.06 mg/dL based compared to a stand-alone RNN model. The difference in prediction accuracy amongst the presented architectures shows that our proposed NPE model is a reliable physiological information encoder. NPE was useful to enhance the prediction power of subsequent sequential networks to make more accurate predictions.

TABLE V

The Prediction Error on OhioT1DM Dataset

| | 559 | 563 | 570 | 575 | 588 | 591 | *RMSE (mg/dL)* |
|---|---|---|---|---|---|---|---|
| RNN | 22.36 | 17.26 | 16.40 | 23.72 | 20.00 | 26.63 | 21.06 |
| GRU | 19.49 | 17.73 | 16.41 | 23.04 | 20.00 | 20.35 | 19.50 |
| LSTM | 18.97 | 17.23 | 16.60 | 23.79 | 19.29 | 21.61 | 19.58 |
| CNN+RNN | 20.00 | 18.16 | 16.27 | 24.17 | 19.26 | 22.15 | 20.00 |
| CNN+GRU | 19.69 | 18.19 | 16.22 | 24.28 | 19.82 | 21.86 | 20.01 |
| CNN+LSTM | 19.19 | 17.79 | 16.24 | 24.24 | 20.09 | 21.99 | 19.92 |
| NPE+RNN | 17.69 | 16.58 | 14.78 | 21.11 | **17.78** | 20.23 | 18.02 |
| NPE+GRU | 17.99 | 17.38 | 15.40 | 21.24 | 18.77 | 22.36 | 18.85 |
| NPE+LSTM | **17.33** | **16.33** | **14.66** | **20.68** | 18.08 | **19.75** | 17.80$^\dagger$ |

Given that the average number of training samples in dataset-2 is 12000 as opposed to 44000 samples in dataset-1, we suspect that this influenced the difference in RMSE between both datasets. A smaller dataset, as with dataset-2, is more difficult to use in training neural network models. In addition, the added dimensions with daily activities associated with this dataset constitutes complex dependencies, which are more difficult to disentangle.

### C. Discussion

Table VI shows a comparison of the results from this paper and others on the task of blood glucose prediction using the publicly available OhioT1DM dataset [22]. This comparison table shows that our proposed NPE model in combination with a downstream sequential neural network model outperforms others in literature. In [13], dilated RNN achieved the RMSE of 18.91 mg/dL by extending the single recurrent layer to multiple layers with different dilation. Meanwhile, in [34], varying configurations of XGBoost was used to achieve an RMSE of 19.32 mg/dL. However, our proposed NPE module can be easily plugged into other downstream networks and trained end-to-end. This approach does not require feature engineering like those in related papers. All NPE combinations outperform existing models, thereby making accurate blood glucose prediction more of a reality. It is important to note that the prediction error generally grows as the prediction horizon increases. For our NPE+LSTM model, the average RMSE steadily increases from around 6 mg/dL to around 30 mg/dL, as prediction horizon was varied from 5 mins to 1 hour. Similar results of around 33 mg/dL for a prediction horizon of 60 mins has been reported in prior literature [35].

## VII. Conclusion

The task of blood glucose prediction is challenging and increasingly important to enable forecasting of adverse glycemic events and support the development of closed-loop artificial pancreas systems. Recent deep learning based models has been used to achieve better prediction accuracy than traditional models such as support vector regression (SVR). In this work, we propose a Neural Physiological

TABLE VI

COMPARISON WITH RELATED WORK ON BLOOD GLUCOSE PREDICTION. PREDICTION HORIZON = 30 MINUTES.

| | Method | RMSE |
|---|---|---|
| Martinson *et al.* [35] | LSTM (trained with NLL) | 20.70 |
| | LSTM (trained with MSE) | 20.10 |
| Zhu *et al.* [25] | WaveNet | 21.73 |
| Chen *et al.* [13] | Dilated RNN | 18.91 |
| Midroni *et al.* [34] | XGBoost (PCA-reduced) | 21.76 |
| | XGBoost (expanded) | 20.37 |
| | XGBoost (feature selection) | 19.32 |
| This work | NPE+RNN | 18.02 |
| | NPE+GRU | 18.85 |
| | NPE+LSTM | 17.80$^\dagger$ |

Encoder (NPE), which can combined with any downstream neural network model and trained end-to-end. A primary advantage of NPE is that no feature engineering is needed during training and prediction process. This is unlike traditional physiological models which require a patient's personal information and professional input to determine the values of mathematical variables. Using two unique datasets, our results show remarkable information-extracting power of NPE over baseline CNN. Regardless of the size of the dataset and the number of the physiological features, NPE can boost the downstream network to generate promising blood glucose prediction. It was shown that NPE can enable improved prediction performance over stand-alone models such as RNN, GRU and LSTM. Additionally, we envision that there is potential to leverage the proposed NPE model for other time-series tasks, such as human activity recognition.

## VIII. ACKNOWLEDGMENT

The authors would like to thank Tidepool for their support and contribution which makes this research possible.

## REFERENCES

[1] M. J. Fowler, "Microvascular and macrovascular complications of diabetes," *Clinical Diabetes*, vol. 26, no. 2, pp. 77–82, 2008.

[2] American Diabetes Association, "Standards of medical care in diabetes," *Diabetes Care*, vol. 40, no. Supplement 1, 2017.

[3] A. Brown, "42 Factors That Affect Blood Glucose," Brown University, Tech. Rep., Accessed: September 2019. [Online]. Available: https://diatribe.org/42factors

[4] S. Morton, R. Li, and T. Prioleau, "Data-driven insights on behavioral factors that affect diabetes management," in *Int. Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2020.

[5] N. Allen and A. Gupta, "Current diabetes technology: striving for the artificial pancreas," *Diagnostics*, vol. 9, no. 1, p. 31, 2019.

[6] B. Danielle, D. Colm, F. Larisa, and D. Dermot, "Glucose sensing for diabetes monitoring: recent developments," *MDPI Sensors*, 2017.

[7] D. Thomas, N. Revital, and B. e. Tadej, "International consensus on use of continuous glucose monitoring," *Diabetes Care*, 2017.

[8] H. Thabit and R. Hovorka, "Coming of age: the artificial pancreas for type 1 diabetes," *Diabetologia*, vol. 59, no. 9, pp. 1795–1805, 2016.

[9] T. Peyser, E. Dassau, M. Breton, and J. S. Skyler, "The artificial pancreas: current status and future prospects in the management of diabetes," *Annals of the New York Academy of Sciences*, vol. 1311, no. 1, pp. 102–123, 2014.

[10] K. Ioannis, T. Olga, S. Athanasios, M. Nicos, V. Ioannis, and C. Ioanna, "Machine learning and data mining methods in diabetes research," *Computational and Structural Biotechnology Journal*, 2017.

[11] K. Plis, R. Bunescu, C. Marling, J. Shubrook, and F. Schwartz, "A machine learning approach to predicting blood glucose levels for diabetes management," in *Workshops at the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[12] N. I. Sapankevych and R. Sankar, "Time series prediction using support vector machines: A survey," *IEEE Comput. Intell. Mag.*, 2009.

[13] J. Chen, K. Li, P. Herrero, T. Zhu, and P. Georgiou, "Dilated recurrent neural network for short-time prediction of glucose concentration." in *KHD@ IJCAI*, 2018, pp. 69–73.

[14] C. Meijner and S. Persson, *Long Short-term Memory based models for blood for blood glucose prediction*. University of Gothenburg, Gothenburg, Sweden, 2017.

[15] M.-L. Clara, D. Robert, T. Nichole, S., R. Navid, and J. Peter, G., "Leveraging a big dataset to develop a recurrent neural network to predict adverse glycemic events in type 1 diabetes," *IEEE Journal of Biomedical and Health Informatics*, 2019.

[16] M. Brook, H. and R. Ali, A., "An overview of insulin pumps and glucose sensors for the generalist," *Journal of Clinical Medicine*, 2016.

[17] P. Kevin, B. Razvan, M. Cindy, S. Jay, and S. Frank, "A machine learning approach to predicting blood glucose levels for diabetes management," in *Assoc. for the Advancement of Artificial Intell.*, 2014.

[18] C. Dalla Man, R. A. Rizza, and C. Cobelli, "Meal simulation model of the glucose-insulin system," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 10, pp. 1740–1749, 2007.

[19] W. Gu, Z. Zhou, Y. Zhou, M. He, H. Zou, and L. Zhang, "Predicting blood glucose dynamics with multi-time-series deep learning," in *ACM Conf. on Embedded Networked Sensor Systems*, 2017.

[20] I. Contreras and J. Vehi, "Artificial intelligence for diabetes management and decision support: literature review," *Journal of Medical Internet Research*, vol. 20, no. 5, p. e10775, 2018.

[21] S. Oviedo, J. Vehí, R. Calm, and J. Armengol, "A review of personalized blood glucose prediction strategies for t1dm patients," *Int. J. Numerical Methods in Biomedical Engineering*, vol. 33, no. 6, 2017.

[22] C. Marling and R. C. Bunescu, "The ohiot1dm dataset for blood glucose level prediction." in *KHD@ IJCAI*, 2018, pp. 60–63.

[23] D. L. Duke, "Intelligent diabetes assistant: A telemedicine system for modeling and managing blood glucose," Ph.D. dissertation, Carnegie Mellon University, Pittsburgh, PA, January 2010.

[24] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[25] T. Zhu, K. Li, P. Herrero, J. Chen, and P. Georgiou, "A deep learning algorithm for personalized blood glucose prediction." in *KHD@ IJCAI*, 2018, pp. 64–78.

[26] K. Li, J. Daniels, C. Liu, P. Herrero-Vinas, and P. Georgiou, "Convolutional recurrent neural networks for glucose prediction," *IEEE journal of biomedical and health informatics*, 2019.

[27] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in *European Converference on Computer Vision*, 2018.

[28] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. on Computer Vision and Pattern Recog.*, 2016, pp. 2818–2826.

[29] Q. Qiu, X. Cheng, R. Calderbank, and G. Sapiro, "Dcfnet: Deep neural network with decomposed convolutional filters," in *35th International Converference on Machine Learning*, 2018.

[30] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conf. on Comp. Vision  Pattern Recog.*, 2015.

[31] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Int. Conf. IEEE Engineering in Medicine  Biology Conference (EMBC)*, 2015.

[32] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Jour. of Machine Learning*, vol. 1, pp. 1929–1958, 2014.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[34] C. Midroni, P. J. Leimbigler, G. Baruah, M. Kolla, A. J. Whitehead, and Y. Fossat, "Predicting glycemia in type 1 diabetes patients: experiments with xgboost," in *KHD@ IJCAI*.

[35] J. Martinsson, A. Schliep, B. Eliasson, C. Meijner, S. Persson, and O. Mogren, "Automatic blood glucose prediction with confidence using recurrent neural networks," in *KHD@ IJCAI*, 2018, pp. 64–68.