Personalized Blood Glucose Prediction for Type 1 Diabetes Using Evidential Deep Learning and Meta-Learning

Taiyu Zhu, *Graduate Student Member, IEEE*, Kezhi Li, *Member, IEEE*, Pau Herrero, *Member, IEEE*, and Pantelis Georgiou, *Senior Member, IEEE*

***Abstract*— The availability of large amounts of data from continuous glucose monitoring (CGM), together with the latest advances in deep learning techniques, have opened the door to a new paradigm of algorithm design for personalized blood glucose (BG) prediction in type 1 diabetes (T1D) with superior performance. However, there are several challenges that prevent the widespread implementation of deep learning algorithms in actual clinical settings, including unclear prediction confidence and limited training data for new T1D subjects. To this end, we propose a novel deep learning framework, Fast-adaptive and Confident Neural Network (FCNN), to meet these clinical challenges. In particular, an attention-based recurrent neural network is used to learn representations from CGM input and forward a weighted sum of hidden states to an evidential output layer, aiming to compute personalized BG predictions with theoretically supported model confidence. The model-agnostic meta-learning is employed to enable fast adaptation for a new T1D subject with limited training data. The proposed framework has been validated on three clinical datasets. In particular, for a dataset including 12 subjects with T1D, FCNN achieved a root mean square error of 18.64±2.60 mg/dL and 31.07±3.62 mg/dL for 30 and 60-minute prediction horizons, respectively, which outperformed all the considered baseline methods with significant improvements. These results indicate that FCNN is a viable and effective approach for predicting BG levels in T1D. The well-trained models can be implemented in smartphone apps to improve glycemic control by enabling proactive actions through real-time glucose alerts.**

***Index Terms*— Artificial intelligence, deep learning, diabetes, glucose prediction, meta-learning**

I. Introduction

DIABETES is a group of metabolic diseases characterized by elevated blood glucose levels (hyperglycemia). It is estimated that almost half a billion people are living with diabetes, and the incidence rates of type 1 diabetes (T1D) and type 2 diabetes are on the rise [1]. T1D is thought to be precipitated by the autoimmune destruction of pancreatic β-cell resulting in an absolute insulin deficiency. People with T1D require lifelong self-management in order to maintain blood glucose (BG) levels in a therapeutically appropriate range. Failure to do so increases the risk of hyperglycemia, which can lead to microvascular and macrovascular complications, and of hypoglycemia, which can lead to coma, or in extreme cases, death [2]. In this context, accurate BG prediction is a valuable tool for enhancing decision support systems in diabetes care, which aim to mitigate these adverse glycemic events and reduce the burdens on people living with diabetes. In particular, BG prediction enables proactive interventions, such as glucose alerts in continuous glucose monitoring (CGM), and the predictive low-glucose basal insulin suspension in currently available CGM-augmented pumps [3].

CGM allows for a real-time tracking of BG levels with a fixed sampling frequency (e.g. every five minutes) and has been verified as an effective device of controlling the blood glucose in T1D management [4], [5]. CGM can be combined with insulin pumps as sensor-augmented therapy or an artificial pancreas to provide closed-loop glycemic control. There is a rising trend of connecting CGM with smartphone apps to display retrospective BG trajectories and allow users to record daily events (e.g. meals, insulin doses, exercise) that have an impact on glucose levels [6]–[8]. The wide use of CGM and mobile apps has produced a large amount of data, which enables the development of data-driven algorithms for personalized BG prediction [9], [10], especially machine learning algorithms [11]–[13]. Among these, empowered by deep neural networks (DNNs), deep learning algorithms have achieved superior performance [14]–[17] with less work on feature engineering [18].

Although deep learning approaches have improved the BG prediction accuracy, there are still many challenges in translating these models into clinical settings. According to feedback from clinicians and clinical outcomes reported in previous studies [13], [19], [20], we identified two clinical challenges that are most significant:

- **Confidence**: A primary concern is whether the provided prediction is reliable enough, and what level of confidence can be given to it (trust issue).
- **Data availability**: Developing a personalized model for a new subject is difficult since DNN models require a large amount of historical data for training, and the data

This work was supported by EPSRC EP/P00993X/1 and President's Ph.D. Scholarship at Imperial College London. *(Corresponding author: K. Li)*

T. Zhu, P. Herrero, P. Georgiou are with Centre for Bio-Inspired Technology, Department of Electrical and Electronic Engineering, Imperial College London, London, United Kingdom. (e-mail: {taiyu.zhu17, pherrero, pantelis}@imperial.ac.uk).

K. Li is with Institute of Health Informatics, University College London, London, United Kingdom. (e-mail: ken.li@ucl.ac.uk).

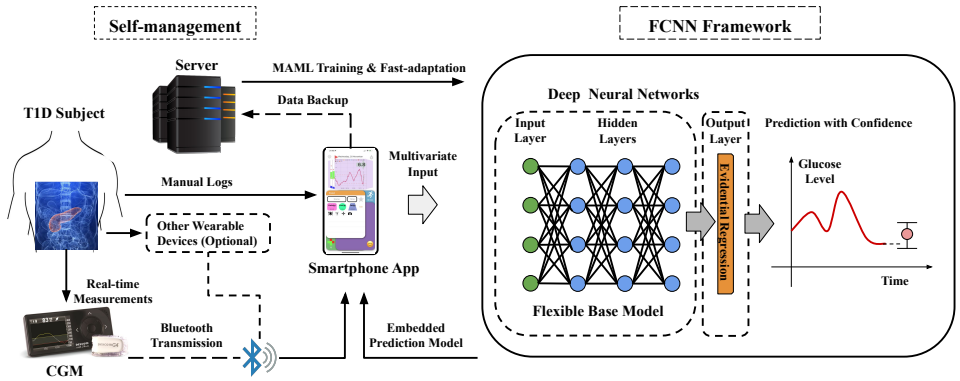

Fig. 1: System architecture of incorporating FCNN in a T1D management system. A smartphone app receives BG measurements from CGM, data from other wearable devices (e.g., insulin pumps and wrist bands), and the records of the relevant daily activities. After uploading the multivariate data to a server (e.g., cloud repository), FCNN is used for fast adaptation of a deep learning model with confidence. Then the well-trained model is embedded into the smartphone app to perform real-time prediction. Besides the bidirectional RNN proposed in this work, FCNN also supports other customized DNNs as base models.

collection can be expensive and time-consuming (cold-start issue).

To this end, we propose a novel deep learning framework for personalized BG prediction, which is called Fast-adaptive and Confident Neural Network (FCNN). Fig. 1 depicts the real-world settings of a T1D management system. Given the multivariate features accessible from wearable devices and a smartphone app, a DNN model can be developed by FCNN framework that incorporates the latest advances in deep learning to address the above challenges, including evidential regression [21] and model-agnostic meta-learning (MAML) [22]. To the best of our knowledge, this is the first work that adopts the evidential deep learning and meta-learning in BG prediction. The experiments show that FCNN exhibits excellent accuracy and outperforms all the considered baseline methods. The main contributions of this article can be summarized as follows.

- We develop a bidirectional recurrent neural network (RNN)-based deep learning model with a modified many-to-one attention mechanism to improve the prediction accuracy and evidential regression that provides theoretically supported confidence intervals, aiming to solve the trust issue.
- We employ MAML with the first-order approximation to enable fast adaptation when developing personalized models with limited data, aiming to solve the cold-start issue.
- We evaluate FCNN on three clinical datasets (Supplementary material S.I), including the OhioT1DM dataset (n = 12), the ARISES Phase 1 dataset (n = 12), and the ABC4D Phase 4 dataset (n = 25), and implement the models in a smartphone app to provide real-time decision support.

The remainder of this article is organized as follows. In Section II, we introduce related work. Section III describes the architecture and components of FCNN for BG prediction. The data description and model performance are presented in Section IV. In the experiments section (Section IV-D and Section IV-E), we first treat MAML as a special case of transfer-learning (TL) to develop personalized models using a two-step data split and all the data of each subject. Then, we conduct a case study in Section IV-F, aiming to determine the efficacy of MAML when only a small amount of data is available for a hold-out subject. We discuss the limitation, future work, and clinical implementation in Section V. Finally, we conclude this article in Section VI.

II. Related Work

Deep learning is a subset of machine learning, which is capable of extracting multiple levels of representation from raw data with multilayer networks, such as RNNs, convolutional neural networks (CNNs), and generative adversarial networks [18]. Particularly, deep learning paradigms have demonstrated promising performance in a number of biomedical and health applications, such as clinical imaging and genomics [20]. As highlighted by a recent review [13], RNN has been widely adopted in BG prediction due to its powerful capability of processing time series data in regression tasks. Unlike plain feed-forward neural networks, RNN units allow for recurrent connections, fetching the output from previous or future timesteps to be used as current input. Vanilla RNNs suffer from gradient vanishing and exploding during back-propagation. To this end, long short-term memory (LSTM) [23] and gated recurrent unit (GRU) [24] were proposed to solve this problem by using a set of element-wise gate functions to control information flow inside the units. Another important improvement for RNNs is the bidirectional connections, which enables the RNN units to simultaneously obtain the information from backward and forward states [25]. These modifications have been shown to be effective approaches at improving the performance of RNN models. For instance, Sun *et al.* [26] introduced an RNN with LSTM and bidirectional LSTM layers for personalized BG prediction, whose performance surpassed traditional machine learning

models (e.g., support vector machine). Similarly, a set of DNN blocks with LSTM layers and residual connections was used in [16] to further enhance prediction accuracy. In [27], the authors proposed a causal GRU-based model with a multitask learning framework to predict BG levels for multiple T1D subjects concurrently.

A recent breakthrough in both natural language processing (NLP) [28] and computer vision [29] is the attention mechanism. In particular, it allows an RNN model to focus on the hidden states and learn long-term temporal dependencies at certain timesteps by calculating a context vector that is weighted by alignment scores. Various attention mechanisms and corresponding score functions were proposed in the recent literature [30], including additive [28], general [31], dot-product [31], scaled dot-product [32], and location-based form [31]. The transformer is based on a multi-head self-attention mechanism, which is a highly successful deep learning architecture in recent NLP studies (e.g., BERT and GPT-3) [33]. In the context of personalized BG prediction, Mirshekarian *et al.* [34] presented a memory-augmented LSTM using the attention weights derived by a two-layer dense network (i.e., additive form), which exhibited improvement on synthetic data. In this article, we adapt another sequence-to-sequence attention mechanism (i.e., general form) for many-to-one prediction (Section III-B), according to model validation performance. Besides, we employ the transformer as a baseline method in the experiments.

In our previous work, we have shown that CNN-based models can also be applied to personalized BG prediction. In [35], we developed a dilated CNN and converted prediction targets into multiple discrete classes. Furthermore, we proposed a hybrid deep neural network, referred to as convolutional recurrent neural network (CRNN), which achieved better performance than the dilated CNN [36]. This model used multiple CNN layers to extract features and then processed the feature representations with an LSTM layer, which is used as a deep learning baseline method for comparison (Section IV).

Taking advantage of historical datasets, TL approaches have been widely used in previous studies to improve personalized BG prediction performance with pretraining and fine-tuning [16], [34], [37]. To further reduce the demand for individual data in the fine-tuning phase, meta-learning, also referent to as learning to learn, is an emerging approach for optimizing a meta-model across a set of learning tasks, so that fast adaptation can be performed with only a few data samples. In particular, MAML [22] can learn the initialization of a DNN model with good average performance and has already been applied to assist clinicians in sleep stage classification [38] but has never been used in the context of BG prediction. Treating each T1D subject as a learning task, MAML is formulated in III-D and compared with a conventional TL method in Section IV-F.

Most of the existing work treated BG prediction as a traditional regression task and used mean square error as the loss function to obtain a single prediction value [13]. In [39], a mixture density network was proposed to model the uncertainty with a univariate Gaussian distribution. However, this configuration can only model data uncertainty by a mixture distribution [40]. Here, we incorporate an evidential layer to simultaneously map data uncertainty and model uncertainty, which is detailed in Section III-C.

III. Methodology

A. Problem Formulation

In general, the input of a data-driven algorithm for BG prediction is multivariate time series, consisting of CGM sequences and other relevant data features (e.g., meals, insulin, exercise), to represent the physiological status of a T1D subject. In this case, the input data $\mathbf{X}_t$ is denoted as

$$\mathbf{X}_t = [\mathbf{x}_{t-L+1}, \mathbf{x}_{t-L+2}, \ldots, \mathbf{x}_t] \in \mathbb{R}^{d\times L}, \tag{1}$$

where $\mathbf{x}_t \in \mathbb{R}^{d\times 1}$ contains d features at the timestep t, and L is the window length, i.e., the number of timesteps of the input. In this work, we selected CGM, meal carbohydrates $\mathbf{M}$, and bolus insulin $\mathbf{I}$, as the input features, which can be defined as $\mathbf{X} = f_N([\mathbf{G}; \mathbf{M}; \mathbf{I}])$, where f_N is the min-max normalization function to scale data within a range of $[0, 1]$. CGM series are obtained from real-time sensor measurements, while the amount of carbohydrates and insulin are based on the daily records from mobile apps.

Given a target prediction horizon (PH) and CGM resolution τ, the BG level at timestep $t + r$ is expressed as G_{t+r}, where $r = \text{PH}/\tau$. To reduce bias [35], [37], [41], we use the signed difference between the current and future BG level as the prediction targets, i.e., $y_t = f_N(G_{t+r} - G_t)$. Hence, the BG prediction $\hat{G}_{t+r}$ can be defined as

$$\hat{G}_{t+r} = f_N^{-1}(\hat{y}_t) + G_t \tag{2}$$

where f_N^{-1} is the inverse function of the normalization, and $\hat{y}_t$ is the output of the deep learning model.

We perform feature preprocessing with the following three steps: removing outliers, imputing missing CGM data, and feature selection. Due to the sparse and random nature of the manually logged events (e.g., meals and insulin doses), which are commonly recorded via a mobile app, we decided to align such manually collected data with the resolution of the CGM measurement (i.e., 5-minute sampling). Outlier detection and removal were performed by applying a set of thresholds based on the maximum and minimum physiological changes between consecutive CGM measurements. Data gaps randomly occur in CGM sequences due to sensor errors, communication problems, and sensor replacement, and usually account for 5% to 10% of the total data. To fill these gaps, we employed a hybrid imputation method. For each input, we linearly interpolated the gaps that occur in the middle of the sequence, and linearly extrapolated the missing samples at the end of the sequence, e.g., $\mathbf{x}_{t-2}, \mathbf{x}_{t-1}, \mathbf{x}_t$, with a bound of $[0, 1]$. Extrapolation was used to ensure that future information is not taken into account when predicting missing measurements. We used high-quality input sequences with missing gaps of less than 15 minutes since linear interpolation is not adequate for imputing data in longer gaps. It should be noted that the preprocessing (i.e., removal) was only performed on the training data but not on validation or testing data.

Supplementary material S.II summarizes the feature set extracted from the clinical datasets, including CGM measurements, bolus insulin, basal insulin, the ingested carbohydrate amount, insulin on board (IOB), carbohydrate on board (COB), and time index series. Bolus insulin is used to compensate for BG increase after meal ingestion and hyperglycemia during fasting, while basal insulin aims at maintaining BG levels in a target range when fasting. IOB and COB, which are commonly employed in bolus calculators and artificial pancreas, were derived from linear models [42] and formulated as follows.

$$\begin{aligned} \text{IOB}_t &= B * \max(0, 1 - \frac{t - t_B}{T_{\text{IOB}}}), \\ \text{COB}_t &= \max(0, C - R_{\text{COB}} * (t - t_C - \Delta_{\text{COB}})), \end{aligned} \tag{3}$$

where B, t_B and C, t_C denote the amount and time of bolus insulin and carbohydrates intake, respectively; T_{IOB} is the active time of insulin effects, and we set it to four hours; R_{COB} is the carbohydrate absorption rate of 0.5 g/min after an initial delay (Δ_{COB}) of 15 minutes. In particular, for the time index, we scaled 24-h time to a range of [0, 1] to encode timesteps that represent time in seconds starting from midnight. After normalizing feature vectors, we performed feature selection for the FCNN model using an exhaustive feature selector and hold-out validation sets. We evaluated performance on the OhioT1DM dataset with an error score es that combines the root mean square error (RMSE) for 30-minute and 60-minute PHs (i.e., $es = (\frac{1}{2}(\text{RMSE}_{30min}^2 + \text{RMSE}_{60min}^2))^{\frac{1}{2}}$ mg/dL). Finally, we selected three input features which led to the best performance, including CGM, amount of bolus insulin, and carbohydrate intake.

B. Model Architecture

Fig. 2 shows the overall structure of the proposed model. We instantiate a stack of three bidirectional GRU layers to extract feature maps from the multivariate input. The complete list of notations can be found in Supplementary material S.III. The computation of a GRU layer, $\mathcal{H}_{\mathbf{W},\mathbf{U},\mathbf{b}}(\mathbf{x}_t, \mathbf{h}')$ in a direction is given by

$$\begin{aligned} \mathbf{z}_t &= \sigma(\mathbf{W}_z \mathbf{x}_t + \mathbf{U}_z \mathbf{h}' + \mathbf{b}_z), \\ \mathbf{r}_t &= \sigma(\mathbf{W}_r \mathbf{x}_t + \mathbf{U}_r \mathbf{h}' + \mathbf{b}_r), \\ \hat{\mathbf{h}}_t &= \sigma(\mathbf{W}_h \mathbf{x}_t + \mathbf{U}_h \mathbf{r}_t \odot \mathbf{h}' + \mathbf{b}_h), \\ \mathbf{h}_t &= (1 - \mathbf{z}_t) \odot \mathbf{h}' + \mathbf{z}_t \odot \hat{\mathbf{h}}_t, \end{aligned} \tag{4}$$

where $\mathbf{z}_t$ and $\mathbf{r}_t$ stand for update gate and reset gate vectors, respectively; σ is the sigmoid function; $\odot$ is the element-wise product; $\mathbf{W}$ and $\mathbf{U}$, and $\mathbf{b}$ are the weights for input, weights for hidden states, and bias, respectively, where the subscripts z, r, and h respectively indicate parameters for the update gate, reset gate, and candidate activation; $\mathbf{h}$, $\mathbf{h}'$, and $\hat{\mathbf{h}}$ are the hidden states of cell output, cell input and candidate activation, respectively. Concatenating backward and forward output, the state output of a bidirectional GRU layer at timestep t is denoted as $\overline{\mathbf{h}}_t = [\overrightarrow{\mathbf{h}}_t; \overleftarrow{\mathbf{h}}_t]$, where $\overline{\mathbf{h}}$ is a concatenated hidden state in bidirectional RNN, which is given by

$$\begin{aligned} \overrightarrow{\mathbf{h}}_t &= \mathcal{H}_{\overrightarrow{\mathbf{W}},\overrightarrow{\mathbf{U}},\overrightarrow{\mathbf{b}}}(\mathbf{x}_t, \mathbf{h}_{t-1}), \\ \overleftarrow{\mathbf{h}}_t &= \mathcal{H}_{\overleftarrow{\mathbf{W}},\overleftarrow{\mathbf{U}},\overleftarrow{\mathbf{b}}}(\mathbf{x}_t, \mathbf{h}_{t+1}). \end{aligned} \tag{5}$$

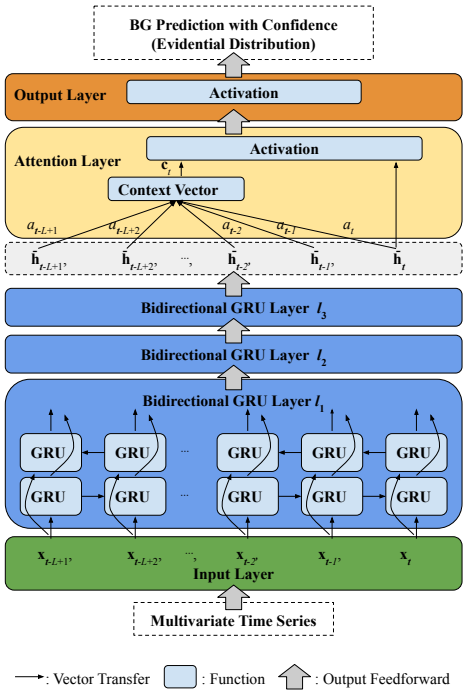

Fig. 2: Diagram of the proposed deep learning model. The input of multivariate time series is first processed by three bidirectional GRU layers to extract feature maps. The attention layer computes a weighted sum of the hidden states. Then the top layer outputs predictions with confidence from an evidential distribution.

where $[\overrightarrow{\mathbf{h}}, \overrightarrow{\mathbf{W}}, \overrightarrow{\mathbf{U}}, \overrightarrow{\mathbf{b}}]$ and $[\overleftarrow{\mathbf{h}}, \overleftarrow{\mathbf{W}}, \overleftarrow{\mathbf{U}}, \overleftarrow{\mathbf{b}}]$ respectively represent the set of output, weights for input, weights for hidden states, bias in forward and backward RNNs.

To enable the DNN model to focus on the important parts of the hidden representations, we introduce a many-to-one attention layer at the top of the GRU layers (Fig. 2). The input of this layer contains a complete sequence of the RNN hidden states. We use a modified general alignment score function [31] to calculate attention weights. Thus, there are a total of two trainable weight matrices ($\mathbf{W}_a$ and $\mathbf{W}_m$) in the attention layer. Each of them is implemented by a dense layer without bias or activation, and therefore it can be updated as a part of the DNN model with gradient descent and backpropagation. The output of the attention layer is an attention vector whose hidden dimension is determined by $\mathbf{W}_m$. Specifically, we first calculate the context vector $\mathbf{c}_t$ as follows

$$\mathbf{c}_t = \sum_{i=t-L+1}^{t} a_i \overline{\mathbf{h}}_i, \tag{6}$$

where $\overline{\mathbf{h}}_i$ is the concatenated hidden state at at the i-th timestep, and the corresponding attention weights a_i is derived by alignment scores.

$$a_i = \frac{\exp(\texttt{score}(\overline{\mathbf{h}}_i, \overline{\mathbf{h}}_t))}{\sum_i \exp(\texttt{score}(\overline{\mathbf{h}}_i, \overline{\mathbf{h}}_t))}. \tag{7}$$

The general alignment score function [31] is formulated as

follows.

$$\texttt{score}(\overline{\mathbf{h}}_i, \overline{\mathbf{h}}_t) = \overline{\mathbf{h}}_t^{\mathsf{T}} \mathbf{W}_a \overline{\mathbf{h}}_i. \tag{8}$$

where $\mathbf{W}_a$ denotes the weights for alignment scores. Given the weighted context vector $\mathbf{c}_t$, the output of the attention middle layer with weights $\mathbf{W}_m$ is defined as

$$\overline{\mathbf{h}}_t = \tanh(\mathbf{W}_m[\mathbf{c}_t; \overline{\mathbf{h}}_t]), \tag{9}$$

which is then fed to an evidential layer for final output.

C. Evidential Deep Learning

The prediction confidence can be estimated by uncertainty levels. In general, there are two types of uncertainty in deep learning: aleatoric uncertainty, i.e., the uncertainty in the data, and epistemic uncertainty, i.e., the uncertainty in the prediction model [40]. Epistemic uncertainty is crucial to determine out-of-distribution shift, indicating whether the prediction can be trusted or not. Bayesian deep learning is a conventional approach to model epistemic uncertainty, but it heavily relies on a complex sampling process during model training [40]. Therefore, inspired by recent advances on evidential deep learning [21], [43], we employ a process of evidence acquisition to simultaneously model aleatoric uncertainty and epistemic uncertainty. Assuming the targets $y_1, y_2, \ldots, y_t$, are the i.i.d. observations following a Gaussian distribution with unknown mean μ and variance σ^2,

$$\mu \sim \mathcal{N}(\gamma, \sigma^2/\lambda), \quad \sigma^2 \sim \Gamma^{-1}(\alpha, \beta), \tag{10}$$

we then have the Gaussian conjugate prior following the normal-inverse-gamma (NIG) distribution: $(\mu, \sigma^2) \sim \texttt{NIG}(\gamma, \lambda, \alpha, \beta)$, as the higher-order evidential distribution. Based on Bayes' theorem, the model evidence $p(y_t|\gamma, \lambda, \alpha, \beta)$, also known as marginal likelihood, can be derived by the likelihood parameters (μ, σ^2) and the probability density function of NIG, following a generalized Student-t distribution St [21]. In this case, to learn the model parameters by maximum likelihood estimation, the corresponding negative log-likelihood (NLL) loss function is given by

$$\begin{aligned} \mathcal{L}_t^N &= -\log p(y_t|\gamma, \lambda, \alpha, \beta) \\ &= -\log(\texttt{St}(y_t|2\alpha, \gamma, \sqrt{\frac{\beta(1+\lambda)}{\lambda\alpha}})), \end{aligned} \tag{11}$$

where 2α is the degrees of freedom; γ and $\sqrt{\frac{\beta(1+\lambda)}{\lambda\alpha}}$ are the location parameter and the scale parameter, respectively. Furthermore, a regularizer is introduced to penalize the errors in the prediction [21],

$$\mathcal{L}_t^R = |y_t - \gamma|(2\lambda + \alpha). \tag{12}$$

Thus, the loss function on t-th sample is

$$\mathcal{L}_t = \mathcal{L}_t^N + k\mathcal{L}_t^R, \tag{13}$$

where k is a hyperparameter to adjust regularization. In this case, the final output of proposed model are $\gamma, \lambda, \alpha, \beta$ in four dimensions. The prediction $\hat{y}_t$, aleatoric uncertainty u_t^a and epistemic uncertainty u_t^e (i.e., prediction uncertainty) are denoted as

$$\hat{y}_t = \gamma, \quad u_t^a = \sqrt{\frac{\beta}{\alpha - 1}}, \quad u_t^e = \sqrt{\frac{\beta}{\lambda(\alpha - 1)}}. \tag{14}$$

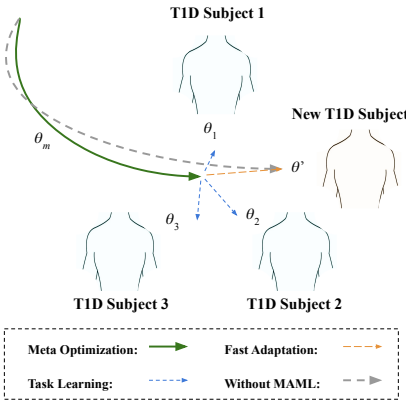

Fig. 3: Illustration of applying meta-learning to BG prediction for fast adaptation. Meta-learning optimizes the model initialization across different subjects in a cohort dataset. Then the initialized model is fast-adaptive to a new subject.

D. Fast adaptation by Meta-learning

Most commercial CGM sensors measure BG levels at intervals of five minutes. Therefore, a maximum of 288 data points can be collected per day. Training a personalized deep learning model usually requires months of clinical data acquisition. Fortunately, datasets collected from historical clinical trials are available for research purposes. To make use of these datasets and accelerate the training process of new personalized models, meta-learning is a feasible approach. Particularly, we employ MAML [22] with the first-order implementation called Reptile [44]. In terms of BG prediction, the learning tasks in MAML can be referred to as predicting BG levels for different T1D subjects in a cohort dataset, as shown in Fig. 3. The meta-models are used in personalized fine-tuning but cannot be used for population-based BG prediction.

The MAML meta-model is expected to minimize loss over a group of different learning tasks (i.e., a T1D cohort). The original MAML relies on an outer loop and an inner loop for meta-optimization and task learning, respectively. Hence, second derivatives are required during the meta-optimization. Calculating a gradient through a gradient is computationally expensive, so a more practical and efficient way to do this is by using the first-order approximation that has the nearly same performance as the original MAML [22]. In this regard, Reptile further simplifies the first-order MAML by reusing the gradients from task learning [44]. Given a task $\mathcal{T}_j$, the j-th update of the model parameters θ_j in the inner loop is defined as

$$\theta_j = \theta_{j-1} - \eta \nabla_{\theta_{j-1}} \mathcal{L}_{B_{\mathcal{T}_j}}(f_{\theta_{j-1}}), \tag{15}$$

where η is the learning rate of an Adam optimizer [45]; $\mathcal{L}_{B_{\mathcal{T}_i}}$ stands for the loss of a mini-batch of data samples from the task $\mathcal{T}_i$; and $f_{\theta_{j-1}}$ is the model inference with parameters θ_{j-1}. Then the update of meta-model parameters θ_m across N tasks,

i.e., the outer loop, is given by

$$\theta_m \leftarrow \theta_m + \frac{\epsilon}{N}\sum_{j=1}^{N}(\theta_j - \theta_m), \tag{16}$$

where ϵ is the step size of stochastic gradient descent. Thus, a pre-trained meta-model with initial parameters θ_m enables fast adaptation and largely reduces the requirement of data availability.

IV. Experiments

A. Clinical Datasets

1) OhioT1DM Dataset: The OhioT1DM dataset is a publicly available dataset [46]. It contains data from 12 subjects with T1D over an eight-week period. All the participants wore a Medtronic Enlite CGM and Medtronic 530G or 630G insulin pumps, and reported their daily events via a smartphone app. Some of them wore Basis Peak or Empatica Embrace bands to collect vital signs data. The data have already been divided into training and testing sets, which account for approximately 80% and 20% of the total samples, respectively.

2) ARISES Dataset: The ARISES dataset is a proprietary dataset (Imperial College London, London, UK) from a six-week clinical trial (NCT03643692) including 12 T1D participants either on multiple daily injection (MDI) or continuous subcutaneous insulin infusion (CSII, i.e., pump therapy). Participants in the trial were equipped with Dexcom G6 CGM and logged daily events via mySugr app. Besides, they used Empatica E4 wrist bands and myTracks app. The study was under the protocol (18/LO/1096) approved by London - Fulham Research Ethics Committee in 2018.

3) ABC4D Dataset: The ABC4D dataset is a proprietary dataset (Imperial College London, London, UK) including data from 25 T1D subjects over six months (NCT02053051) [47]. The dataset contains BG levels measured with Dexcom G5 CGM and multiple self-reported events, including meal ingestion, physical exercise, and basal-bolus insulin regimens. The study was under the protocol (13/LO/0264) approved by London - Chelsea Research Ethics Committee in 2013.

B. Model Configurations

We performed a two-step hold-out data split (Supplementary material S.IV). The training and testing sets of the OhioT1DM dataset are provided separately. Thus, we first divided the data of each subject in the ARISES and ABC4D datasets into a training set including the first 80% of the data and a hold-out testing set including the remaining 20%. Secondly, we split each training set of the OhioT1DM, ARISES and ABC4D datasets again into an actual training set with the first 80% of the data and a hold-out validation set with the rest 20%. In this way, the training, validation, and testing sets respectively contain 64%, 16%, and 20% of the full data of the ARISES and the ABC4D datasets. The models were trained using the actual training sets, and validation and testing sets were unseen data. Then the validations sets were used for feature selection and hyperparameter tuning. Finally, the testing sets were used to provide unbiased evaluation and the prediction results reported in this work. We strictly followed chronological partitions to split time series data to avoid data leakage and guaranteed that the testing and validation sets did not include any data from the training sets. Similar split methods have been widely used in previous studies on BG prediction [27], [34], [35], [37].

The chosen values for the hyperparameters are listed in Supplementary material S.V. Early stopping was used to mitigate overfitting and improve the generalization of the DNN models, for which we set the total number of epochs to 500 with the patience of 50. We compared the performance of the proposed model with several classic data-driven baseline methods in the literature [13]. Support vector regression (SVR) [48], random forest regression (RFR) [49], and autoregressive integrated moving average (ARIMA) [50] were used as classic machine learning baselines, while bidirectional LSTM (Bi-LSTM) [26] and a variant of the transformer [32] were used as deep learning baselines. The original transformer is a sequence-to-sequence model, so we adopted its encoder as the prediction model [51]. We also used a CRNN model as a baseline method, considering it is the best model in our previous work [36]. We tuned these baseline models using the same hold-out validation sets. The input features of the baseline models are the same as those of FCNN, including CGM, carbohydrate, and bolus insulin, except for ARIMA which only uses CGM data.

We developed the FCNN models and the other DNNs with Python 3.8, TensorFlow 2.3, and Keras 2.4 with a GPU acceleration (NVIDIA GTX 1080 Ti). The classic machine learning methods were developed with scikit-learn 0.23 (SVR, RFR) and statsmodels 0.12 (ARIMA). Finally, We conducted the experiments with the PHs of 30 and 60 minutes. These PHs are commonly employed since they allow timely intervention to prevent undesired glycemic events [13].

C. Evaluation Metrics

We used commonly employed evaluation metrics in BG prediction: RMSE and mean absolute error (MAE) with the detailed values of percentages in the A, B, C, D and E regions of the Clark error grid (CEG) [52]. To evaluate the clinical performance, we also used the glucose-specific RMSE (gRMSE) [53] and prediction time delay (PTD) [35]–[37] for a comprehensive evaluation. The definition of the metrics is presented in Supplementary material S.VI.

D. Results

Table I, II and III respectively summarize the prediction results for the OhioT1DM dataset, the ARISES dataset and the ABC4D dataset with PH of 30 and 60 minutes (Mean$\pm$STD). Notably, for both PHs, FCNN achieved the best performance on each dataset with the smallest RMSE, MAE, gRMSE, and PTD. Moreover, after evaluating the normality of the results with Shapiro–Wilk tests, we performed paired t-tests to indicate statistical significance with respect to the proposed baseline methods. The FCNN obtains significant improvements on RMSE results for each dataset, but the improvements on gRMSE results are less significant, especially for the

TABLE I: Prediction performance of the considered prediction methods on the OhioT1DM dataset

Methods	RMSE (mg/dL)	MAE (mg/dL)	gRMSE (mg/dL)	PTD (min)	CEG-Regions (%)				
					A	B	C	D	E
PH = 30 minutes									
FCNN	$\mathbf{18.64 \pm 2.60}$	$\mathbf{13.25 \pm 1.67}$	$\mathbf{22.86 \pm 3.47}$	$\mathbf{6.75 \pm 4.30}$	$\mathbf{89.80 \pm 3.65}$	8.96 ± 2.85	0.01 ± 0.02	1.22 ± 0.93	0.01 ± 0.01
CRNN	$19.38 \pm 2.39^{\dagger}$	$13.79 \pm 1.64^{*}$	$24.12 \pm 3.14^{\dagger}$	7.42 ± 4.66	$88.72 \pm 3.90^{*}$	$9.72 \pm 2.98^{*}$	0.02 ± 0.03	$1.55 \pm 1.14^{*}$	0.00 ± 0.00
Bi-LSTM	$19.59 \pm 2.22^{\ddagger}$	$13.79 \pm 1.31^{\dagger}$	$24.30 \pm 3.14^{\ddagger}$	6.82 ± 4.08	$88.93 \pm 3.07^{*}$	$9.74 \pm 2.49^{\ddagger}$	0.02 ± 0.03	1.30 ± 0.84	0.01 ± 0.02
Transformer	$19.69 \pm 2.36^{\ddagger}$	$13.90 \pm 1.42^{\ddagger}$	$24.60 \pm 3.25^{\ddagger}$	7.56 ± 4.49	$89.15 \pm 3.46^{*}$	$9.77 \pm 2.80^{\ddagger}$	0.01 ± 0.03	1.07 ± 0.71	0.00 ± 0.01
SVR	$21.10 \pm 2.31^{\ddagger}$	$15.98 \pm 1.94^{\ddagger}$	$26.49 \pm 3.14^{\ddagger}$	$7.94 \pm 3.76^{*}$	$82.70 \pm 6.05^{\ddagger}$	$14.58 \pm 4.32^{\ddagger}$	0.01 ± 0.02	$2.71 \pm 2.23^{*}$	0.00 ± 0.01
RFR	$21.18 \pm 2.26^{\ddagger}$	$15.30 \pm 1.61^{\ddagger}$	$26.33 \pm 2.97^{\ddagger}$	$8.25 \pm 3.86^{*}$	$86.71 \pm 4.48^{\ddagger}$	$11.70 \pm 3.52^{\ddagger}$	0.02 ± 0.04	$1.56 \pm 1.10^{*}$	0.00 ± 0.01
ARIMA	$20.39 \pm 2.21^{\ddagger}$	$14.40 \pm 1.41^{\ddagger}$	$24.48 \pm 2.68^{\ddagger}$	$11.36 \pm 4.43^{\ddagger}$	$88.41 \pm 3.48^{\ddagger}$	$10.94 \pm 3.19^{\ddagger}$	0.03 ± 0.06	$0.61 \pm 0.38^{*}$	0.01 ± 0.01
PH = 60 minutes									
FCNN	$\mathbf{31.07 \pm 3.62}$	$\mathbf{22.86 \pm 2.89}$	$\mathbf{39.78 \pm 5.28}$	$\mathbf{14.58 \pm 9.91}$	$\mathbf{72.58 \pm 7.87}$	24.39 ± 6.41	0.16 ± 0.14	2.85 ± 1.68	0.02 ± 0.04
CRNN	$32.02 \pm 3.76^{\ddagger}$	$23.82 \pm 3.13^{\ddagger}$	$41.25 \pm 5.37^{\ddagger}$	16.01 ± 10.03	$71.06 \pm 8.69^{*}$	25.57 ± 7.07	0.15 ± 0.17	$3.20 \pm 1.99^{*}$	0.01 ± 0.04
Bi-LSTM	$33.44 \pm 3.76^{\ddagger}$	$24.59 \pm 2.89^{\ddagger}$	$43.45 \pm 5.42^{\ddagger}$	16.71 ± 8.82	$70.61 \pm 8.21^{\dagger}$	$25.98 \pm 6.70^{*}$	0.17 ± 0.13	3.19 ± 1.89	0.05 ± 0.07
Transformer	$32.96 \pm 3.70^{\ddagger}$	$24.19 \pm 2.79^{\ddagger}$	$42.82 \pm 5.22^{\dagger}$	14.81 ± 10.66	71.70 ± 7.77	25.20 ± 6.43	0.15 ± 0.15	2.92 ± 1.65	0.04 ± 0.05
SVR	$33.83 \pm 3.62^{\ddagger}$	$25.63 \pm 2.98^{\ddagger}$	$43.88 \pm 5.07^{\ddagger}$	$24.00 \pm 9.67^{\ddagger}$	$66.43 \pm 9.15^{\ddagger}$	$29.61 \pm 7.30^{\dagger}$	0.20 ± 0.21	$3.73 \pm 2.62^{*}$	0.03 ± 0.04
RFR	$35.31 \pm 3.72^{\ddagger}$	$26.43 \pm 3.02^{\ddagger}$	$45.32 \pm 5.13^{\ddagger}$	$23.10 \pm 10.61^{\dagger}$	$67.03 \pm 8.17^{\ddagger}$	$29.38 \pm 6.29^{\ddagger}$	0.23 ± 0.19	$3.34 \pm 2.14^{*}$	0.02 ± 0.04
ARIMA	$35.42 \pm 3.74^{\ddagger}$	$25.97 \pm 2.70^{\ddagger}$	$43.78 \pm 4.68^{\ddagger}$	$35.12 \pm 10.58^{\ddagger}$	$68.77 \pm 6.85^{*}$	$28.65 \pm 5.83^{\ddagger}$	$0.46 \pm 0.40^{*}$	$2.06 \pm 1.00^{*}$	0.05 ± 0.05

$^{*}p \leq 0.05$ $^{\dagger}p \leq 0.01$ $^{\ddagger}p \leq 0.005$.

TABLE II: Prediction performance of the considered prediction methods on the ARISES dataset

Methods	RMSE (mg/dL)	MAE (mg/dL)	gRMSE (mg/dL)	PTD (min)	CEG-Regions (%)				
					A	B	C	D	E
PH = 30 minutes									
FCNN	$\mathbf{20.23 \pm 3.38}$	$\mathbf{14.67 \pm 2.36}$	$\mathbf{25.20 \pm 4.52}$	$\mathbf{8.36 \pm 4.22}$	$\mathbf{87.24 \pm 5.86}$	10.60 ± 4.20	0.02 ± 0.07	2.14 ± 1.75	0.00 ± 0.00
CRNN	$20.76 \pm 3.71^{\ddagger}$	$15.02 \pm 2.62^{*}$	$26.14 \pm 4.98^{\dagger}$	8.64 ± 4.73	86.92 ± 6.12	10.79 ± 4.48	0.03 ± 0.05	2.27 ± 1.81	0.00 ± 0.00
Bi-LSTM	$20.95 \pm 3.11^{\ddagger}$	$15.29 \pm 2.22^{\ddagger}$	$26.54 \pm 4.13^{*}$	8.49 ± 4.48	86.63 ± 6.21	10.92 ± 4.38	0.02 ± 0.04	2.43 ± 1.92	0.00 ± 0.00
Transformer	$22.45 \pm 4.08^{\ddagger}$	$16.38 \pm 3.01^{\ddagger}$	$28.73 \pm 5.54^{\ddagger}$	$10.24 \pm 5.36^{\ddagger}$	$85.29 \pm 7.50^{*}$	$12.37 \pm 5.53^{\ddagger}$	0.04 ± 0.09	2.30 ± 2.03	0.00 ± 0.00
SVR	$22.26 \pm 4.21^{\ddagger}$	$16.85 \pm 3.20^{\ddagger}$	$28.19 \pm 5.85^{\ddagger}$	9.52 ± 4.82	$82.57 \pm 8.57^{\ddagger}$	$14.45 \pm 6.37^{\ddagger}$	0.03 ± 0.05	2.94 ± 2.35	0.00 ± 0.00
RFR	$23.86 \pm 4.44^{\ddagger}$	$17.56 \pm 3.18^{\ddagger}$	$30.14 \pm 5.99^{\ddagger}$	$10.28 \pm 5.21^{\ddagger}$	$83.22 \pm 7.19^{\ddagger}$	$14.24 \pm 5.44^{\ddagger}$	0.07 ± 0.11	2.47 ± 1.85	0.00 ± 0.00
ARIMA	$21.75 \pm 4.02^{\ddagger}$	$15.59 \pm 2.70^{\ddagger}$	$26.21 \pm 5.23^{*}$	$11.56 \pm 4.89^{\ddagger}$	86.43 ± 6.07	$12.68 \pm 5.42^{\ddagger}$	0.04 ± 0.06	$0.85 \pm 0.72^{\ddagger}$	0.00 ± 0.00
PH = 60 minutes									
FCNN	$\mathbf{35.40 \pm 7.04}$	$\mathbf{26.23 \pm 5.00}$	$\mathbf{45.93 \pm 9.60}$	$\mathbf{24.44 \pm 11.05}$	$\mathbf{69.29 \pm 8.91}$	26.27 ± 6.51	0.33 ± 0.46	4.11 ± 2.60	0.01 ± 0.02
CRNN	$36.08 \pm 6.99^{*}$	$26.86 \pm 5.14^{\ddagger}$	$47.58 \pm 10.03^{\ddagger}$	24.84 ± 11.24	68.52 ± 9.56	26.99 ± 7.15	0.24 ± 0.28	4.24 ± 2.63	0.01 ± 0.03
Bi-LSTM	$36.83 \pm 7.27^{\ddagger}$	$27.44 \pm 5.38^{\ddagger}$	$48.13 \pm 10.09^{\ddagger}$	25.53 ± 11.48	$67.50 \pm 9.51^{\ddagger}$	$28.00 \pm 7.09^{\ddagger}$	0.30 ± 0.39	4.18 ± 2.61	0.01 ± 0.03
Transformer	$36.98 \pm 6.96^{\ddagger}$	$27.59 \pm 5.27^{\ddagger}$	$48.66 \pm 10.01^{\ddagger}$	26.14 ± 12.30	$67.14 \pm 9.57^{\ddagger}$	$28.33 \pm 7.11^{\ddagger}$	0.21 ± 0.29	4.30 ± 2.66	0.02 ± 0.04
SVR	$37.06 \pm 7.55^{\ddagger}$	$27.94 \pm 5.66^{\ddagger}$	$48.74 \pm 10.71^{\ddagger}$	27.32 ± 12.17	$66.42 \pm 10.32^{\dagger}$	$28.94 \pm 7.88^{*}$	0.31 ± 0.38	4.30 ± 2.80	0.03 ± 0.06
RFR	$39.46 \pm 7.73^{\ddagger}$	$29.78 \pm 5.87^{\ddagger}$	$51.58 \pm 10.79^{\ddagger}$	26.69 ± 11.91	$64.34 \pm 9.51^{\ddagger}$	$30.92 \pm 7.20^{\ddagger}$	0.41 ± 0.43	4.24 ± 2.54	$0.09 \pm 0.12^{*}$
ARIMA	$39.46 \pm 8.13^{\ddagger}$	$28.71 \pm 5.62^{\ddagger}$	$49.70 \pm 10.97^{\ddagger}$	$36.00 \pm 12.44^{\ddagger}$	$67.05 \pm 9.52^{\ddagger}$	$29.19 \pm 7.76^{\ddagger}$	$0.56 \pm 0.53^{\ddagger}$	$2.91 \pm 1.65^{\dagger}$	$0.29 \pm 0.34^{*}$

$^{*}p \leq 0.05$ $^{\dagger}p \leq 0.01$ $^{\ddagger}p \leq 0.005$.

TABLE III: Prediction performance of the considered prediction methods on the ABC4D dataset

Methods	RMSE (mg/dL)	MAE (mg/dL)	gRMSE (mg/dL)	PTD (min)	CEG-Regions (%)				
					A	B	C	D	E
PH = 30 minutes									
FCNN	$\mathbf{20.25 \pm 2.60}$	$\mathbf{14.50 \pm 1.95}$	$\mathbf{25.00 \pm 3.49}$	$\mathbf{7.76 \pm 3.79}$	$\mathbf{86.50 \pm 3.90}$	11.54 ± 2.84	0.03 ± 0.06	1.93 ± 1.20	0.00 ± 0.01
CRNN	$20.55 \pm 2.55^{*}$	$14.77 \pm 1.86^{\dagger}$	$25.44 \pm 3.46^{*}$	8.23 ± 4.62	$85.94 \pm 3.75^{\dagger}$	11.78 ± 2.46	0.03 ± 0.06	$2.25 \pm 1.47^{\ddagger}$	0.00 ± 0.00
Bi-LSTM	$20.66 \pm 2.45^{\ddagger}$	$14.85 \pm 1.77^{\ddagger}$	$25.68 \pm 3.22^{\ddagger}$	7.90 ± 4.42	$85.72 \pm 4.15^{\dagger}$	11.81 ± 2.74	0.03 ± 0.06	$2.43 \pm 1.72^{\ddagger}$	0.00 ± 0.02
Transformer	$20.63 \pm 2.66^{\dagger}$	$14.84 \pm 1.91^{\ddagger}$	$25.60 \pm 3.59^{\ddagger}$	8.87 ± 4.85	$85.89 \pm 3.98^{*}$	11.92 ± 2.68	0.03 ± 0.06	2.17 ± 1.56	0.00 ± 0.00
SVR	$21.90 \pm 2.44^{\ddagger}$	$16.73 \pm 1.85^{\ddagger}$	$27.70 \pm 3.45^{\ddagger}$	8.71 ± 4.65	$79.87 \pm 6.55^{\ddagger}$	$15.59 \pm 3.28^{\ddagger}$	0.03 ± 0.06	$4.51 \pm 4.30^{\ddagger}$	0.00 ± 0.00
RFR	$21.80 \pm 2.48^{\ddagger}$	$15.83 \pm 1.81^{\ddagger}$	$27.12 \pm 3.34^{\ddagger}$	8.69 ± 4.63	$83.91 \pm 4.30^{\ddagger}$	$13.37 \pm 2.65^{\ddagger}$	0.03 ± 0.06	$2.68 \pm 1.93^{\ddagger}$	0.00 ± 0.01
ARIMA	$22.13 \pm 2.60^{\ddagger}$	$15.60 \pm 1.90^{\ddagger}$	$26.48 \pm 3.56^{\ddagger}$	$13.55 \pm 4.65^{\ddagger}$	$85.11 \pm 3.81^{\ddagger}$	$13.67 \pm 3.18^{\ddagger}$	$0.06 \pm 0.06^{\ddagger}$	$1.14 \pm 0.77^{\ddagger}$	$0.01 \pm 0.02^{\ddagger}$
PH = 60 minutes									
FCNN	$\mathbf{34.03 \pm 4.74}$	$\mathbf{25.26 \pm 3.60}$	$\mathbf{44.06 \pm 6.25}$	$\mathbf{21.47 \pm 7.59}$	$\mathbf{68.01 \pm 5.37}$	27.07 ± 3.47	0.23 ± 0.24	4.67 ± 2.87	0.03 ± 0.04
CRNN	$34.39 \pm 4.75^{\ddagger}$	$25.63 \pm 3.53^{\ddagger}$	$44.58 \pm 6.31^{\dagger}$	$25.56 \pm 10.94^{\dagger}$	$67.07 \pm 5.53^{\ddagger}$	$27.72 \pm 3.29^{*}$	0.24 ± 0.29	4.95 ± 3.35	0.02 ± 0.05
Bi-LSTM	$34.82 \pm 4.44^{\dagger}$	$25.86 \pm 3.27^{*}$	$45.17 \pm 5.98^{*}$	22.78 ± 9.78	$66.71 \pm 5.89^{*}$	$28.13 \pm 3.56^{*}$	0.25 ± 0.25	4.88 ± 3.35	0.03 ± 0.05
Transformer	$34.91 \pm 4.43^{\dagger}$	$25.99 \pm 3.28^{*}$	$45.42 \pm 6.13^{\dagger}$	$26.08 \pm 11.65^{*}$	$66.68 \pm 5.89^{*}$	28.26 ± 3.88	0.22 ± 0.26	4.82 ± 3.12	0.02 ± 0.04
SVR	$35.37 \pm 4.70^{\ddagger}$	$26.98 \pm 3.50^{\ddagger}$	$45.99 \pm 6.43^{\ddagger}$	$26.99 \pm 11.30^{\ddagger}$	$63.62 \pm 7.01^{\ddagger}$	$30.44 \pm 4.04^{\ddagger}$	$0.28 \pm 0.29^{*}$	$5.62 \pm 4.54^{*}$	0.03 ± 0.05
RFR	$36.57 \pm 4.76^{\ddagger}$	$27.43 \pm 3.55^{\ddagger}$	$47.27 \pm 6.40^{\ddagger}$	$24.95 \pm 10.28^{*}$	$64.60 \pm 6.22^{\ddagger}$	$30.02 \pm 3.76^{\ddagger}$	$0.36 \pm 0.36^{\ddagger}$	4.98 ± 3.63	0.04 ± 0.06
ARIMA	$38.55 \pm 5.15^{\ddagger}$	$28.00 \pm 3.75^{\ddagger}$	$47.94 \pm 7.16^{\ddagger}$	$39.42 \pm 7.48^{\ddagger}$	$65.22 \pm 4.88^{\ddagger}$	$30.94 \pm 3.63^{\ddagger}$	$0.59 \pm 0.46^{\ddagger}$	$3.05 \pm 1.55^{\ddagger}$	$0.20 \pm 0.15^{\ddagger}$

$^{*}p \leq 0.05$ $^{\dagger}p \leq 0.01$ $^{\ddagger}p \leq 0.005$.

ABC4D dataset. The reason for this reduced performance with the ABC4D dataset might be explained by the large number of missing data points in several subjects. Supplementary material S.VII presents the results of mean error (ME), where we observe that the FCNN achieved smaller ME and thus lower bias in the estimates when compared with the baseline methods on the three datasets. Thus, the proposed method exhibited good average performance in terms of all the mean errors, but its impact on the clinical benefits needs to be further improved. This is the reason why we introduced model

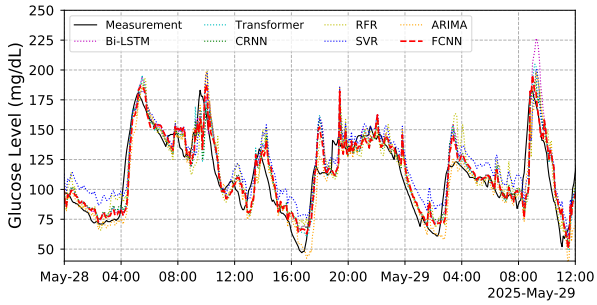

Fig. 4: 1.5-day period forecasting performance of the considered methods over the 30-minute PH for the OhioT1DM dataset. The solid black line indicates BG levels measured by CGM, and the dashed red line indicates the prediction results of FCNN. The magenta, cyan, green, yellow, blue, and orange lines respectively indicate the baselines of Bi-LSTM, Transformer, CRNN, RFR, SVR, and ARIMA.

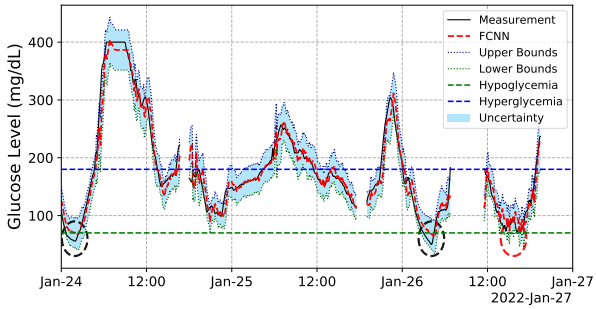

Fig. 5: Forecasting results corresponding to the proposed method including prediction confidence on a subject in the OhioT1DM dataset. The solid black line indicates CGM measurements, and the dashed red line indicates the prediction results of FCNN. The dotted blue and green lines respectively indicate the upper and lower bounds derived by evidential regression. The shaded light blue area indicates the confidence interval. The dashed blue and green lines respectively indicate the thresholds of hyperglycemia and hypoglycemia.

confidence to predict adverse glycemic events with adjustable lower and upper bounds (Section IV-E).

Overall, the accuracy of BG prediction decreased as the PHs became larger. The daily events that might occur within the 60-minute period (e.g. meals, corrections boluses, exercise) have an affect on BG levels and make the prediction more challenging. It is worth noting that all the three deep learning methods outperformed the classic machine learning baselines, demonstrating the good learning behaviors of DNNs. Regarding the RMSE performance, the transformer is comparable to the Bi-LSTM, which exhibits better results with the 60-minute PH in the OhioT1DM dataset, and with the 30-minute PH in the ABC4D dataset.

Fig. 4 shows the trajectories of the BG predictions on a subject in the OhioT1DM dataset. Compared with the baseline methods, we see that the FCNN performed well, especially at the peaks and troughs of the BG curve. That is, the errors in the hypoglycemia and hyperglycemia regions are small (e.g., the predictions around 9:00 on May 29). Similar performance can be observed in the ABC4D dataset and the ARISES dataset. These findings on the BG trajectories are consistent with the results in the above tables. Supplementary material S.VII depicts the CEG analysis for FCNN, which is evaluated on three T1D subjects and two PHs. We observed that the dots concentrate on the A and B regions. These two regions include the predictions that are within 20% error with respect to the actual CGM measurements, and would not lead to inappropriate treatment. The corresponding numerical results of each region are presented in Table I, II and III. It is to be noted that, when evaluated on the OhioT1DM dataset, 98.86% of FCNN predictions for the 30-minute PH are located within the A and B regions.

E. Prediction Confidence

Although FCNN achieves substantial improvements in these regions, a significant delay can still be observed in Fig. 4. To address this clinical challenge, we incorporated the evidential deep learning and modeled uncertainty defined in Equation (14). Fig. 5 depicts an instance of using the lower bounds of the confidence interval (CI) to successfully identify two hypoglycemia regions in the dashed black ellipses, which are likely to be missed using single prediction values. Considering epistemic uncertainty interprets the confidence of predictions, we calculated upper and lower bounds as $[\hat{G}_{t+r} - f_N^{-1}(u_t^e), \hat{G}_{t+r} + f_N^{-1}(u_t^e)]$.

TABLE IV: Performance of hypoglycemia prediction methods on the OhioT1DM dataset

Method	Sensitivity (%)	Precision (%)	FPPD	MCC
PH = 30 minutes				
FCNN	84.09 ± 5.73	65.60 ± 17.68	6.07 ± 4.30	0.72 ± 0.10
CRNN	$44.34 \pm 25.53^{\ddagger}$	70.03 ± 23.33	1.88 ± 1.43	$0.53 \pm 0.23^{\ddagger}$
Bi-LSTM	$45.12 \pm 23.13^{\ddagger}$	56.58 ± 19.93	4.71 ± 3.50	$0.48 \pm 0.20^{\ddagger}$
Transformer	$59.35 \pm 18.82^{\ddagger}$	71.36 ± 16.48	3.69 ± 1.94	$0.61 \pm 0.15^{\ddagger}$
SVR	$12.84 \pm 11.73^{\ddagger}$	54.84 ± 36.37	$1.12 \pm 1.04^{\ddagger}$	$0.24 \pm 0.18^{\ddagger}$
RFR	$47.86 \pm 25.16^{\ddagger}$	62.49 ± 21.08	3.21 ± 2.21	$0.52 \pm 0.21^{\ddagger}$
ARIMA	86.32 ± 6.04	$50.94 \pm 9.61^{*}$	10.87 ± 4.74	0.63 ± 0.06
PH = 60 minutes				
FCNN	68.58 ± 14.39	60.64 ± 18.82	14.45 ± 15.86	0.59 ± 0.08
CRNN	$15.83 \pm 13.30^{\ddagger}$	42.58 ± 29.52	2.60 ± 2.27	$0.23 \pm 0.18^{\ddagger}$
Bi-LSTM	$28.73 \pm 19.32^{\ddagger}$	52.77 ± 26.34	5.77 ± 3.78	$0.34 \pm 0.17^{\ddagger}$
Transformer	$38.71 \pm 27.98^{*}$	53.92 ± 29.71	5.81 ± 4.94	$0.40 \pm 0.23^{*}$
SVR	$9.46 \pm 8.91^{\ddagger}$	$34.29 \pm 30.45^{*}$	$3.25 \pm 2.95^{*}$	$0.15 \pm 0.14^{\ddagger}$
RFR	$19.03 \pm 24.44^{\ddagger}$	32.63 ± 30.99	2.70 ± 3.02	$0.22 \pm 0.19^{\ddagger}$
ARIMA	80.17 ± 8.99	43.80 ± 9.51	19.55 ± 9.33	0.54 ± 0.07

$^{*}p \leq 0.05$ $^{\dagger}p \leq 0.01$ $^{\ddagger}p \leq 0.005$.

Hypoglycemia is more dangerous than hyperglycemia, since it can lead to acute coma or even death in severe cases [54]. To quantify the effectiveness of the prediction confidence, we altered the bounds by $[\hat{G}_{t+r} - zf_N^{-1}(u_t^e), \hat{G}_{t+r} + zf_N^{-1}(u_t^e)]$, where z is the ratio of the uncertainty and treated as a hyperparameter with a range of [0, 1] in this work. We evaluated the results using four classification metrics (Supplementary material S.VI): sensitivity, precision, false-positives per day (FPPD), and Matthews correlation coefficient (MCC). Considering that hypoglycemia is a minority class, accounting for 3-5% of whole BG trajectories (Supplementary material S.I), we use FPPD as an alternative way to present the specificity. It indicates how often the algorithm would lead to false hy-

poglycemia alarms in decision support systems or closed-loop systems. Rescue treatment for impending hypoglycemia, such as rescue carbohydrates, is commonly performed in T1D management. Hence, a hypoglycemic event was considered when there was a single BG level below 70 mg/dL. Table IV presents the hypoglycemia prediction performance of FCNN and the baseline methods. It is noted that, FCNN outperformed all the considered baseline methods with higher MCC scores for the 30-minute and 60-minute PHs. Although ARIMA exhibited the large RMSE and PTD results for BG prediction across the three datasets, it obtained good performance for hypoglycemia prediction. A possible explanation is that ARIMA performed well at the troughs of BG curves but showed significant delay in other regions (Fig. 4). It is reasonable that FCNN yields high FPPD due to the use of the lower bounds for detecting hypoglycemia. However, most false positives occurred around actual hypoglycemic events. If each event triggers a single alarm, FCNN can still achieve a small number of false alarms per day (Supplementary material S.VI) of 0.48 ± 0.53 for the 60-minute PH.

There is a trade-off between precision and sensitivity, while high MCC scores can be obtained only if model performs well on all the confusion matrix categories. To avoid hypoglycemia, higher sensitivity is preferred in clinical settings at the cost of slightly less precision (i.e., increase of false positive rate), as indicated by the dashed black and red ellipses in Fig. 5, but too low precision might cause alarm fatigue. In this regard, the ratio z is flexible and can be chosen by clinicians.

F. Adaptation Performance

A common use case of fast adaptation is supposed to be fine-tuning a meta-model for a new T1D subject with an increasing amount of data, starting from a very small batch of available data. In this section, we present a case study assuming that only the first 14 days of training data are available for a hold-out target subject in a cohort dataset, aiming to test the day-to-day performance of fast adaptation. The training data of the other subjects remain unchanged for the development of MAML meta-models. We chose a length of 14 days because the lifespan of most commercial CGM sensors in clinical settings is between 7 and 14 days.

For comparison purposes, a pretrained model with TL techniques was employed as a baseline, which has been commonly used in the literature of BG prediction [16], [34], [37], [55]. The TL-pretrained model has the same DNN architecture as that of FCNN, and developing such a model includes two steps. First, we excluded the data of the target subject and combined the training data of the remaining subjects to form a global set. Then, we pretrained a model with the mini-batch data randomly sampled from the global set, and each input batch corresponds to a single subject.

After performing MAML and TL without the hold-out data, we fine-tuned the meta-model and TL-pretrained model using the same individual data of the hold-out target subject. In order to simulate the 14-day use case across a lifespan of a CGM sensor, we sequentially added one-day data into the individual data set and repeated the fine-tuning experiments on each experimental day. Each time we used the same testing sets as those in the previous Section IV-D for evaluation. The experiments were repeated for all the subjects.

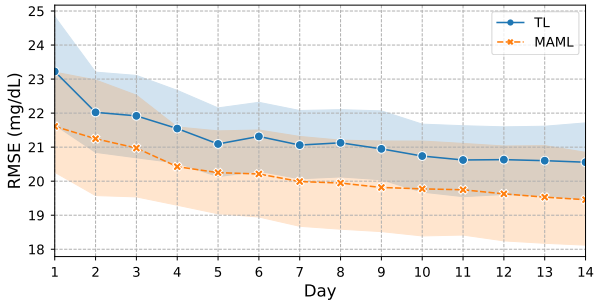

Fig. 6: 30-minute RMSE results of fine-tuning on the FCNN meta-model and the TL-pretrained model with the small-size training sets of the OhioT1DM dataset. The solid blue and orange lines are the mean RMSE of TL and MAML, respectively, and shaded areas stand for 95% CI.

TABLE V: 30-minute RMSE of the fast adaptation methods across the three datasets

Day	Method	OhioT1DM	ARISES	ABC4D
1	MAML	21.61 ± 2.59	25.66 ± 5.26	22.87 ± 2.81
	TL	23.22 ± 4.27	27.05 ± 7.03	24.32 ± 3.78
7	MAML	19.99 ± 2.35	23.00 ± 3.95	22.54 ± 3.16
	TL	21.06 ± 2.56	23.78 ± 3.65	22.94 ± 2.60
14	MAML	$\mathbf{19.46 \pm 2.48}$	$\mathbf{21.00 \pm 3.62}$	$\mathbf{21.53 \pm 2.53}$
	TL	20.56 ± 2.81	22.02 ± 3.78	22.30 ± 2.66

Fig. 6 depicts the performance of FCNN and the TL baseline on the OhioT1DM dataset. Table V summarizes the results for the three datasets. In this limit case, when compared with TL, MAML provided an average RMSE improvement of 1.48 and 0.96 mg/dL on day 1 and day 14, respectively. The meta-models by MAML exhibited better prediction performance during the whole fine-tuning process and achieved much smaller RMSE from the start of the fine-tuning (day 1), when the size of available data was extremely small. These findings are consistent with a recent study on MAML, which has proven that the effectiveness of MAML is primarily due to feature reuse [56]. That is, the meta-initialized models were already good at learning representations for a new subject. MAML achieved an RMSE below 20 on day 7, while it took much longer for the TL-pretrained models to reach this level. We further performed the experiments using the data of the first 25 days, which is the maximum mutual length of the data in the training sets. It is observed that the RMSE of MAML keep decreasing until day 18 for the OhioT1DM and ARISES datasets, and day 20 for the ABC4D dataset, and then become stable.

In the general case, where enough training data were available for each subject, the use of MAML significantly reduced RMSE by 0.2 mg/dL ($p < 0.005$, Supplementary material S.VIII). These results indicate that the use of MAML is a feasible approach to enable fast adaptation and improve model performance with a small size of available data. It should be noted that the MAML meta-models are not the population models for BG prediction, which cannot be directly evaluated

on the testing data of new subjects. Here we only indicate that MAML outperformed TL during the fine-tuning of the development of personalized models.

V. Discussion

To the best of our knowledge, FCNN is the first work that uses an attention-based GRU model for BG prediction, while incorporating model confidence and fast adaptation for the clinical benefits. There are several significant differences between the attention-based LSTM model presented by Mirshekarian *et al.* [34] and the proposed model in this work. Firstly, the authors evaluated the model on a previous version of the OhioT1DM dataset (2018 version), which contains six of the 12 T1D subjects in the 2020 OhioT1DM dataset that we used. If we evaluate our models on the six 2018 subjects, a 30-minute RMSE of 18.10 mg/dL can be obtained, which is lower than the best result of 18.70 mg/dL that the authors reported using the same experimental settings (i.e., agnostic scenarios without what-if events). Secondly, the authors used an additive form of attention module [28] to obtain a hidden state of LSTM cells with the largest attention weight, while we use a general form of attention mechanism [31] to calculate the weighted sum of the hidden states of bidirectional GRU cells. Finally, their attention module did not improve BG prediction for real clinical data, while ours significantly reduced RMSE by 0.45 mg/dL ($p < 0.005$) in ablation analysis (Supplementary material S.VIII).

In the experiments, we have shown that FCNN is a superior prediction method when compared with the chosen baseline algorithms. In general, it is difficult to perform a fair comparison with the existing work, due to unavailable code, data, and experimental settings. A recent work proposed the glucose variability impact index (GVII) and glucose prediction consistency index (GPCI) as a method to assess the correlation between RMSE results of BG prediction and glucose variability [57], which can be used to compare algorithms across different studies. Here we measured glucose variability by means of the coefficient of variation (CV) and applied linear least-squares regression to obtain GVII and GPCI for each prediction method. However, it is to be noted that the correlation results on the OhioT1DM and ARISES dataset are not significant, possibly due to the small numbers of T1D subjects. Thus, we reported the GVII and GPCI results with Pearson correlation coefficients (r) and p-values (p) on the ABC4D dataset in Supplementary material S.VII. It is worth noting that the FCNN method achieved small GVII and GPCI results for both PHs, indicating that glucose variability has a low impact on the accuracy and consistency of BG prediction. Moreover, the FCNN framework can be adapted to many existing DNN models to improve their performance, such as the CNN [35], the CRNN [36] and the dilated RNN [37]. Particularly, such adaptation only involves three steps: replacing the dense top layer with evidential output layer, using the corresponding NLL loss, and applying the MAML procedures.

A limitation of FCNN is that the model outputs, i.e. the predictive BG levels, sometimes increase with an input of bolus insulin. Although it is reasonable when there is an upcoming meal, it would be more appropriate for a model to disentangle the effect of meal intake and insulin delivery before it can be used in clinical settings. This limitation is also found in other deep learning models, including the CRNN, Bi-LSTM and the transformer. To this end, there are two potential solutions for future work. One is to introduce monotonic constraints in the DNNs to specify the insulin's negative effect on BG levels, such as restricting the layer weights of shallow networks [58] and training with heuristic regularizations [59]. The other is to incorporate physiological models to process these events, such as composite minimal models of glucose regulation [60]. Real-time hypoglycemia detection based on BG prediction is a challenging task. It is to be noted that, in the literature, the sensitivity of 75% and precision of 51% [61], and the sensitivity of 59% and precision of 68% [55], were obtained for a 30-minute PH, while an MCC score of 0.51 with the sensitivity of 48.5% was achieved for a 60-minute PH [62]. Therefore, the performance of FCNN (Table IV) is better than the state of the art. A potential improvement can be made by introducing another regularizer into the loss function to penalize the error of hypoglycemia detection (Equation (13)), although it may result in a larger RMSE for BG prediction. Meanwhile, we noted that the normalized time index used in this work lacks continuity for the time around midnight. Thus, we will explore sine and cosine embeddings to model time sequences in the future [16]. We are also planning to include other data features known to influence BG levels [63], such as the vital signs measured by wrist bands in the OhioT1DM and ARISES datasets, as well as the biomarkers that can be easily derived from available measurements (e.g., heart rate variability [64]). In addition, developing a population model to enable BG prediction [57] and hypoglycemia prediction [65] for a new T1D subject without model fine-tuning or personalized data is an interesting and relevant future work to be considered. However, a dataset with a much larger sample size such as the Tidepool Big Data Donation Dataset [57], may be required to capture the effects of inter-subject variability. Finally, FCNN can be used as an encoder in deep reinforcement learning to extract hidden features from physiological environment and improve decision support systems and automated insulin delivery algorithms (e.g., artificial pancreas) [66]–[68].

To translate the potential clinical benefits of FCNN to people living with T1D, we are planning to evaluate it in an actual clinical setting, following the system architecture displayed in Fig. 1. A fast-adaptive meta-model is trained by using a historic dataset, for which the personalized fine-tuning will be performed with the newly collected data in the upcoming trial. Although weeks or months of data are required to achieve the reported performance in Table I, II and III, it is not unusual for CGM users to have such an amount of data collected [69]. As shown in Supplementary material S.IX, we implement the FCNN model in an iOS platform to provide real-time predictions and display the upper and lower bounds in the app interface to indicate the risk of adverse glycemic events. To test the on-device performance, we converted the trained model to a mobile-compatible format

using TensorFlow Lite. It turns out that FCNN only consumes 1.15 MB of storage memory with an average execution time of 4 ms for model inference, which is proven to be a feasible approach on a mobile platform to provide real-time glucose alerts. In future work, the FCNN model with only CGM input will also be investigated with smartphone implementation and edge devices [70] to provide sub-optimal BG prediction for T1D subjects whose meal and insulin data are not available.

VI. Conclusion

In this work, we propose a novel deep learning framework called FCNN to overcome the crucial clinical challenges in personalized BG prediction: confidence in the predictions and data availability. Incorporating the meta-learning and evidential deep learning, we developed a fast-adaptive and confident deep learning model based on the bidirectional GRU and attention mechanism. We have shown that FCNN significantly outperformed the selected baseline methods, in terms of RMSE, MAE, and gRMSE, when evaluated on the three clinical datasets. The confidence bounds derived by evidential regression can notably improved MCC scores of hypoglycemia detection, especially for the 60-minute PH. Compared with the classic TL baseline, the use of MAML obtained superior performance of fast-adaptation when training data are limited. The FCNN models can be integrated in glycemic control with smartphone-based implementation, which has great potential to enhance T1D self-management in clinical settings.

Acknowledgement

This research has been funded by Engineering and Physical Sciences Research Council (EPSRC EP/P00993X/1) and the President's PhD Scholarship at Imperial College London (UK). We would like to thank Prof. Nick Oliver, Dr. Monika Reddy, Dr. Chukwuma Uduku, and Narvada Jugnee for their contribution in obtaining the ABC4D and ARISES datasets.

References

[1] P. Saeedi, I. Petersohn, P. Salpea, B. Malanda, S. Karuranga, N. Unwin, S. Colagiuri, L. Guariguata, A. A. Motala, K. Ogurtsova, and Others, "Global and regional diabetes prevalence estimates for 2019 and projections for 2030 and 2045: Results from the International Diabetes Federation Diabetes Atlas," *Diabetes Research and Clinical Practice*, vol. 157, p. 107843, 2019.

[2] E. W. Gregg, N. Sattar, and M. K. Ali, "The changing face of diabetes complications," *The Lancet Diabetes & Endocrinology*, vol. 4, no. 6, pp. 537–547, 2016.

[3] T. Battelino, R. Nimri, K. Dovc, M. Phillip, and N. Bratina, "Prevention of hypoglycemia with predictive low glucose insulin suspension in children with type 1 diabetes: A randomized controlled trial," *Diabetes Care*, vol. 40, no. 6, pp. 764–770, 2017.

[4] Juvenile Diabetes Research Foundation Continuous Glucose Monitoring Study Group, "Continuous glucose monitoring and intensive treatment of type 1 diabetes," *New England Journal of Medicine*, vol. 359, no. 14, pp. 1464–1476, 2008.

[5] D. Rodbard, "Continuous glucose monitoring: A review of successes, challenges, and opportunities," *Diabetes Technology & Therapeutics*, vol. 18, no. S2, pp. S2—-3, 2016.

[6] J. A. Cafazzo, M. Casselman, N. Hamming, D. K. Katzman, and M. R. Palmert, "Design of an mHealth app for the self-management of adolescent type 1 diabetes: A pilot study," *Journal of Medical Internet Research*, vol. 14, no. 3, p. e70, 2012.

[7] M. Kirwan, C. Vandelanotte, A. Fenning, and M. J. Duncan, "Diabetes self-management smartphone application for adults with type 1 diabetes: Randomized controlled trial," *Journal of Medical Internet Research*, vol. 15, no. 11, p. e235, 2013.

[8] P. Keith-Hynes, B. Mize, A. Robert, and J. Place, "The diabetes assistant: a smartphone-based system for real-time control of blood glucose," *Electronics*, vol. 3, no. 4, pp. 609–623, 2014.

[9] 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*, vol. 54, no. 5, pp. 931–937, 2007.

[10] S. Oviedo, J. Vehí, R. Calm, and J. Armengol, "A review of personalized blood glucose prediction strategies for T1DM patients," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 33, no. 6, p. e2833, 2017.

[11] J. Xie and Q. Wang, "Benchmarking machine learning algorithms on blood glucose prediction for type i diabetes in comparison with classical time-series models," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 11, pp. 3101–3124, 2020.

[12] A. Z. Woldaregay, E. Årsand, S. Walderhaug, D. Albers, L. Mamykina, T. Botsis, and G. Hartvigsen, "Data-driven modeling and prediction of blood glucose dynamics: Machine learning applications in type 1 diabetes," *Artificial Intelligence in Medicine*, 2019.

[13] T. Zhu, K. Li, P. Herrero, and P. Georgiou, "Deep learning for diabetes: A systematic review," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 7, pp. 2744–2757, 2021.

[14] T. Zhu, K. Li, P. Herrero, J. Chen, and P. Georgiou, "A deep learning algorithm for personalized blood glucose prediction," in *The 3rd International Workshop on Knowledge Discovery in Healthcare Data, IJCAI-ECAI 2018*, 2018, pp. 74–78.

[15] J. Chen, K. Li, P. Herrero, T. Zhu, and P. Georgiou, "Dilated recurrent neural network for short-time prediction of glucose concentration," in *The 3rd International Workshop on Knowledge Discovery in Healthcare Data, IJCAI-ECAI 2018*, 2018, pp. 69–73.

[16] H. Rubin-Falcone, I. Fox, and J. Wiens, "Deep residual time-series forecasting: Application to blood glucose prediction," in *The 5th International Workshop on Knowledge Discovery in Healthcare Data, ECAI 2020*, 2020, pp. 105–109.

[17] E. M. Aiello, G. Lisanti, L. Magni, M. Musci, and C. Toffanin, "Therapy-driven deep glucose forecasting," *Engineering Applications of Artificial Intelligence*, vol. 87, p. 103255, 2020.

[18] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[19] D. Ravì, C. Wong, F. Deligianni, M. Berthelot, J. Andreu-Perez, B. Lo, and G.-Z. Yang, "Deep learning for health informatics," *IEEE Journal of Biomedical and Health Informatics*, vol. 21, no. 1, pp. 4–21, 2016.

[20] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, "Deep learning for healthcare: Review, opportunities and challenges," *Briefings in Bioinformatics*, vol. 19, no. 6, pp. 1236–1246, 2018.

[21] A. Amini, W. Schwarting, A. Soleimany, and D. Rus, "Deep evidential regression," in *Advances in Neural Information Processing Systems*, 2020.

[22] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1126–1135.

[23] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[24] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *NIPS 2014 Workshop on Deep Learning, December 2014*, 2014.

[25] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[26] Q. Sun, M. V. Jankovic, L. Bally, and S. 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)*. IEEE, 2018, pp. 1–5.

[27] M. He, W. Gu, Y. Kong, L. Zhang, C. J. Spanos, and K. M. Mosalam, "CausalBG: Causal recurrent neural network for the blood glucose inference with IoT platform," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 598–610, 2019.

[28] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations*, 2015.

[29] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation

with visual attention," in *International Conference on Machine Learning*. PMLR, 2015, pp. 2048–2057.

[30] A. Galassi, M. Lippi, and P. Torroni, "Attention in natural language processing," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[31] M. T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Conference Proceedings - EMNLP 2015: Conference on Empirical Methods in Natural Language Processing*, 2015.

[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[33] T. Wolf, J. Chaumond, L. Debut, V. Sanh, C. Delangue, A. Moi, P. Cistac, M. Funtowicz, J. Davison, S. Shleifer *et al.*, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2020, pp. 38–45.

[34] S. Mirshekarian, H. Shen, R. Bunescu, and C. Marling, "LSTMs and neural attention models for blood glucose prediction: Comparative experiments on real and synthetic data," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2019, pp. 706–712.

[35] K. Li, C. Liu, T. Zhu, P. Herrero, and P. Georgiou, "GluNet: A deep learning framework for accurate glucose forecasting," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 2, pp. 414–423, 2020.

[36] K. Li, J. Daniels, C. Liu, P. Herrero, and P. Georgiou, "Convolutional recurrent neural networks for glucose prediction," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 2, pp. 603–613, 2020.

[37] T. Zhu, K. Li, J. Chen, P. Herrero, and P. Georgiou, *Journal of Healthcare Informatics Research*, vol. 4, no. 3, pp. 308–324, 2020.

[38] N. Banluesombatkul, P. Ouppaphan, P. Leelaarporn, P. Lakhan, B. Chaitusaney, N. Jaimchariya, E. Chuangsuwanich, W. Chen, H. Phan, N. Dilokthanakul *et al.*, "Metasleeplearner: A pilot study on fast adaptation of bio-signals-based sleep stage classifier to new individual subject using meta-learning," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 6, pp. 1949–1963, 2021.

[39] J. Martinsson, A. Schliep, B. Eliasson, and O. Mogren, "Blood glucose prediction with variance estimation using recurrent neural networks," *Journal of Healthcare Informatics Research*, vol. 4, no. 1, pp. 1–18, 2020.

[40] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 5580–5590.

[41] T. Zhu, X. Yao, K. Li, P. Herrero, and P. Georgiou, "Blood glucose prediction for type 1 diabetes using generative adversarial networks," in *The 5th International Workshop on Knowledge Discovery in Healthcare Data, ECAI 2020*, 2020, pp. 90–94.

[42] D. Dave, D. J. DeSalvo, B. Haridas, S. McKay, A. Shenoy, C. J. Koh, M. Lawley, and M. Erraguntla, "Feature-based machine learning model for real-time hypoglycemia prediction," *Journal of Diabetes Science and Technology*, vol. 15, no. 4, pp. 842–855, 2021.

[43] M. Sensoy, L. Kaplan, and M. Kandemir, "Evidential deep learning to quantify classification uncertainty," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 3183–3193.

[44] "On first-order meta-learning algorithms," *CoRR*, vol. abs/1803.02999, 2018. [Online]. Available: http://arxiv.org/abs/1803.02999

[45] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, pp. 1–15, 2015.

[46] C. Marling and R. Bunescu, "The OhioT1DM dataset for blood glucose level prediction: Update 2020," in *The 5th KDH workshop, ECAI 2020*, 2020, pp. 71–74.

[47] P. Herrero, A. Alalitei, M. Reddy, P. Georgiou, and N. Oliver, "Robust determination of the optimal continuous glucose monitoring length of intervention to evaluate long-term glycaemic control," *Diabetes Technology and Therapeutics*, no. ja, 2020.

[48] E. I. Georga, V. C. Protopappas, D. Polyzos, and D. I. Fotiadis, "A predictive model of subcutaneous glucose concentration in type 1 diabetes based on random forests," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2012, pp. 2889–2892.

[49] E. I. Georga, V. C. Protopappas, D. Ardigò, M. Marina, I. Zavaroni, D. Polyzos, and D. I. Fotiadis, "Multivariate prediction of subcutaneous glucose concentration in type 1 diabetes patients based on support vector regression," *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 1, pp. 71–81, 2013.

[50] 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.

[51] L. Yang, T. L. J. Ng, B. Smyth, and R. Dong, "Html: Hierarchical transformer-based multi-task learning for volatility prediction," in *Proceedings of The Web Conference 2020*, 2020, pp. 441–451.

[52] W. L. Clarke, D. Cox, L. A. Gonder-Frederick, W. Carter, and S. L. Pohl, "Evaluating clinical accuracy of systems for self-monitoring of blood glucose," *Diabetes Care*, vol. 10, no. 5, pp. 622–628, 1987.

[53] S. Del Favero, A. Facchinetti, and C. Cobelli, "A glucose-specific metric to assess predictors and identify models," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 5, pp. 1281–1290, 2012.

[54] J.-F. Yale, B. Paty, and P. A. Senior, "Hypoglycemia," *Canadian Journal of Diabetes*, vol. 42, pp. S104—-S108, 2018.

[55] Y. Deng, L. Lu, L. Aponte, A. M. Angelidi, V. Novak, G. E. Karniadakis, and C. S. Mantzoros, "Deep transfer learning and data augmentation improve glucose levels prediction in type 2 diabetes patients," *npj Digital Medicine*, vol. 4, no. 1, pp. 1–13, 2021.

[56] A. Raghu, M. Raghu, S. Bengio, and O. Vinyals, "Rapid learning or feature reuse? towards understanding the effectiveness of maml," in *International Conference on Learning Representations*, 2019.

[57] C. Mosquera-Lopez and P. G. Jacobs, "Incorporating glucose variability into glucose forecasting accuracy assessment using the new glucose variability impact index and the prediction consistency index: An LSTM case example," *Journal of Diabetes Science and Technology*, vol. 16, no. 1, pp. 7–18, 2022.

[58] T. Kushner, M. D. Breton, and S. Sankaranarayanan, "Multi-hour blood glucose prediction in type 1 diabetes: A patient-specific approach using shallow neural network models," *Diabetes Technology & Therapeutics*, vol. 22, no. 12, pp. 883–891, 2020.

[59] X. Liu, X. Han, N. Zhang, and Q. Liu, "Certified monotonic neural networks," in *Advances in Neural Information Processing Systems*, 2020.

[60] C. Liu, J. Vehí, P. Avari, M. Reddy, N. Oliver, P. Georgiou, and P. Herrero, "Long-term glucose forecasting using a physiological model and deconvolution of the continuous glucose monitoring signal," *Sensors*, vol. 19, no. 19, p. 4338, 2019.

[61] M. Gadaleta, A. Facchinetti, E. Grisan, and M. Rossi, "Prediction of adverse glycemic events from continuous glucose monitoring signal," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 2, pp. 650–659, 2019.

[62] J. Vehí, I. Contreras, S. Oviedo, L. Biagi, and A. Bertachi, "Prediction and prevention of hypoglycaemic events in type-1 diabetic patients using machine learning," *Health Informatics Journal*, vol. 26, no. 1, pp. 703–718, 2020.

[63] M. Sevil, M. Rashid, I. Hajizadeh, M. Park, L. Quinn, and A. Cinar, "Physical activity and psychological stress detection and assessment of their effects on glucose concentration predictions in diabetes management," *IEEE Transactions on Biomedical Engineering*, 2021.

[64] T. Zhu, C. Uduku, K. Li, P. Herrero, N. Oliver, and P. Georgiou, "Enhancing self-management in type 1 diabetes with wearables and deep learning," *npj Digital Medicine*, vol. 5, no. 1, p. 78, 2022.

[65] N. S. Tyler, C. Mosquera-Lopez, G. M. Young, J. El Youssef, J. R. Castle, and P. G. Jacobs, "Quantifying the impact of physical activity on future glucose trends using machine learning," *iScience*, p. 103888, 2022.

[66] T. Zhu, K. Li, and P. Georgiou, "Personalized dual-hormone control for type 1 diabetes using deep reinforcement learning," in *International Workshop on Health Intelligence (W3PHIAI-20) in the 34th AAAI Conference on Artificial Intelligence*, 2020.

[67] T. Zhu, K. Li, L. Kuang, P. Herrero, and P. Georgiou, "An insulin bolus advisor for type 1 diabetes using deep reinforcement learning," *Sensors*, vol. 20, no. 18, p. 5058, 2020.

[68] T. Zhu, K. Li, P. Herrero, and P. Georgiou, "Basal glucose control in type 1 diabetes using deep reinforcement learning: An in silico validation," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 4, pp. 1223–1232, 2021.

[69] J. Walsh, R. Roberts, D. Weber, G. Faber-Heinemann, and L. Heinemann, "Insulin pump and CGM usage in the United States and Germany: results of a real-world survey with 985 subjects," *Journal of Diabetes Science and Technology*, vol. 9, no. 5, pp. 1103–1110, 2015.

[70] T. Zhu, L. Kuang, J. Daniels, P. Herrero, K. Li, and P. Georgiou, "IoMT-enabled real-time blood glucose prediction with deep learning and edge computing," *IEEE Internet of Things Journal*, pp. 1–1, 2022.