

SugarMate: Non-intrusive Blood Glucose Monitoring with Smartphones

WEIXI GU, Tsinghua-Berkeley Shenzhen Institute, Tsinghua University

YUXUN ZHOU, EECS, University of California, Berkeley

ZIMU ZHOU, Computer Engineering and Networks Laboratory, ETH Zurich

XI LIU, Tsinghua-Berkeley Shenzhen Institute, Tsinghua University

HAN ZOU, EECS, University of California, Berkeley

PEI ZHANG, ECE, Carnegie Mellon University

COSTAS J. SPANOS, EECS, University of California, Berkeley

LIN ZHANG, Tsinghua-Berkeley Shenzhen Institute, Tsinghua University

Inferring abnormal glucose events such as hyperglycemia and hypoglycemia is crucial for the health of both diabetic patients and non-diabetic people. However, regular blood glucose monitoring can be invasive and inconvenient in everyday life. We present SugarMate, a first smartphone-based blood glucose inference system as a temporary alternative to continuous blood glucose monitors (CGM) when they are uncomfortable or inconvenient to wear. In addition to the records of food, drug and insulin intake, it leverages smartphone sensors to measure physical activities and sleep quality automatically. Provided with the imbalanced and often limited measurements, a challenge of SugarMate is the inference of blood glucose levels at a fine-grained time resolution. We propose Md³RNN, an efficient learning paradigm to make full use of the available blood glucose information. Specifically, the newly designed grouped input layers, together with the adoption of a deep RNN model, offer an opportunity to build blood glucose models for the general public based on limited personal measurements from single-user and grouped-users perspectives. Evaluations on 112 users demonstrate that Md³RNN yields an average accuracy of 82.14%, significantly outperforming previous learning methods those are either shallow, generically structured, or oblivious to grouped behaviors. Also, a user study with the 112 participants shows that SugarMate is acceptable for practical usage.

CCS Concepts: •Human-centered computing → Smartphones; •Computing methodologies → Machine learning;

ACM Reference format:

Weixi Gu, Yuxun Zhou, Zimu Zhou, Xi Liu, Han Zou, Pei Zhang, Costas J. Spanos, and Lin Zhang. 2017. SugarMate: Non-intrusive Blood Glucose Monitoring with Smartphones. *PACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 3, Article 54 (September 2017), 27 pages.

DOI: <http://doi.org/10.1145/3130919>

1 INTRODUCTION

Blood glucose concentration plays an important role in personal health. Hyperglycemia (high blood glucose level) results in diabetes, leading to health risks such as pancreatic function failure, immunity reduction and ocular

Authors address: Weixi Gu, Xi Liu and Lin Zhang, Tsinghua-Berkeley Shenzhen Institute, Shenzhen, China; E-mails: guweixigavin@gmail.com, liu.xi@sz.tsinghua.edu.cn, linzhang@tsinghua.edu.cn; Authors address: Yuxun Zhou, Han Zou and Costas J. Spanos, Electrical Engineering and Computer Sciences, University of California, Berkeley, USA; E-mails: fyxzhou@berkeley.edu, hanzou@berkeley.edu, spanos@berkeley.edu; Authors address: Zimu Zhou, Computer Engineering and Networks Laboratory, ETH Zurich, Switzerland; E-mail: zzhou@tik.ee.ethz.ch; Authors address: Pei Zhang, Department of Electrical and Computer Engineering, Carnegie Mellon University, USA; E-mail: peizhang@cmu.edu;

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2017 ACM. 2474-9567/2017/9-ART54 \$15
DOI: <http://doi.org/10.1145/3130919>

fundus diseases [27]. Meanwhile, hypoglycemia (low blood glucose level) also brings complications such as confusion, shakiness, anxiety, and even coma or death if not treated in time [20]. People with diabetes need tight control of their blood glucose concentration to avoid both short-term and long-term physiological complications. While non-diabetic people normally have adequate self-regulation of blood glucose concentration, they are still exposed to the risk of hypoglycemia when taking prolonged exercises, drinking excess amounts of alcohol, having eating disorders, taking certain medicines (e.g., certain painkiller and antibiotic), or having pre-diabetes [11] [13].

Although continuous blood glucose monitoring is crucial for blood glucose management and beneficial for hyper- and hypoglycemia warnings, it can be invasive and inconvenient, especially during daily life. A standard blood glucose measurement is to collect and analyze a drop of blood by finger pricking, which requires a new prick on the finger for every observation. Alternatively, non-invasive (without penetrating the skin) continuous glucose monitoring (CGM) has attracted extensive research leveraging techniques such as thermal infrared spectroscopy, raman spectroscopy and impedance spectroscopy [51]. However, most CGM devices are expensive, cumbersome to wear for extended time, and complicated in terms of operation/maintenance, making them unattractive for both diabetic patients and non-diabetic people.

Towards more ubiquitous blood glucose monitoring when traditional CGM devices are unavailable or inconvenient to wear, researchers have proposed exploring the increasingly rich sensors embedded in commercial fitness wearables and smartphones as a complement. In addition to the inherent glucose metabolism, blood glucose also correlates to easily measurable physiological activities such as food and drug intake, energy expenditure, sleep quality and emotional states [24]. Pioneer works [37] [50] have proposed preliminary systems adopting bio-sensors (e.g., ECG electrodes) and fitness bands (e.g., accelerometer and galvanic skin response sensors) to predict blood glucose concentrations and alarm abnormal blood glucose events. Nevertheless, they are validated with a limited number of measurements and for short-term (e.g., 3 hours without the need of CGM devices [50]) or still require complex multi-sensory platforms [37] [50].

We design SugarMate, a personalized smartphone-based non-invasive blood glucose monitoring system that detects abnormal blood glucose events by jointly tracking meal, drug and insulin intake, as well as physical activity and sleep quality. SugarMate is designed as a *temporary* alternative for CGM devices when they are uncomfortable or inconvenient to wear. It monitors blood glucose level every 3 minutes and can be used for three weeks before re-calibration by the CGM devices¹. The blood glucose levels tracked by SugarMate help both diabetic patients and non-diabetic people who want to track their blood glucose to adjust their lifestyle as needed (e.g., exercise more and take less carbohydrate) and take other precaution measures when an abnormal blood glucose level is detected. SugarMate considers *generic*, *grouped* and *user-specific* correlations between blood glucose levels and the measurable external factors, which are largely overlooked in previous works.

The key challenge in designing SugarMate is to learn effective, accurate and personalized blood glucose models. While there have been general blood glucose models characterizing universal trends between blood glucose concentration and various external factors [41], they have to be adjusted based on user-specific data to account for inter-user differences [3]. Yet it is often difficult to collect sufficient data to directly build up personalized models [35]: (i) A disposable enzyme of glucose sensor embedded in the CGM device is only capable of a few days [55] [34], and most users are unwilling to wear CGM devices frequently due to discomfort. (ii) Despite their importance, measurements of hyper- and hypoglycemia events are rare compared to normal blood glucose concentrations, making it difficult to accurately detect abnormal blood glucose events.

To take full advantages of the *sparse*, *imbalanced* measurements to build *personalized* blood glucose level models, we first conduct feature extraction from both physiological and temporal perspectives. We propose Md³RNN (multi-division deep dynamic recurrent neural network), an efficient learning paradigm that extracts general blood glucose level relevant features and preserves user-specific characteristics. Md³RNN advances

¹A one time CGM calibration usually lasts six days, which is a normal disposable usage duration of the enzyme in the sensor of the CGM.

previous personalized recurrent neural network (RNN) structures via a group-shared input layer to extract distinctive feature representations within the same group (*i.e.*, non-diabetic, type I and type II diabetic). In short, Md³RNN can be regarded as a deep hierarchical RNN architecture, which is a combination of single user division and grouped user division learning. Evaluations on the blood glucose dataset composed of 112 users collected for 6 to 30 days show that Md³RNN outperforms both generic learning (*i.e.*, ignoring inter-user differences) and personalized learning (due to lack of measurements), and also achieves notably higher inference accuracy than conventional shallow learning algorithms.

The key contributions of this work are summarized as follows.

- To the best of our knowledge, SugarMate is the first smartphone-based personalized blood glucose monitor that works without CGM devices for an extended duration of time. It automatically collects daily exercise and sleep quality, and infers the current blood glucose level of users, together with manual records of food, drug and insulin intake. It only needs re-calibration using CGM devices once every three weeks.
- We propose Md³RNN, an efficient multi-division deep dynamic RNN framework for blood glucose level inference. The newly designed grouped input layers, together with the adoption of a deep RNN model, offer an opportunity to build blood glucose models for the general public based on limited personal measurements from single-user and grouped-users views.
- We conduct extensive evaluations and user studies on both diabetic patients and non-diabetic people. Experimental results from a dataset of 35 non-diabetic people, and 38 type I and 39 type II diabetic patients collected for at least 6 days per person demonstrate that SugarMate yields an average accuracy of 82.14%, and outperforms traditional general learning, group-level learning, personalized learning and shallow/deep learning algorithms in precision and recall. The user study conducted with these 112 participants shows that most participants are willing to adopt SugarMate as a temporary alternative to CGM for the blood glucose monitoring, and they can also get instructions on their blood glucose control.

In the rest of this paper, we review related works in Sec. 2 and present an overview of SugarMate in Sec. 3. The design and evaluation of SugarMate are detailed in Sec. 4 and Sec. 5, respectively. We conduct a user study in Sec. 6. Finally, we conclude in Sec. 7.

2 RELATED WORK

SugarMate is related to the following categories of research.

2.1 Physiological Models and Blood Glucose Prediction

Physiological models [8] [23] mathematically formulate the whole process of glucose metabolism and are widely used for simulations and studies for blood glucose prediction. Research on blood glucose prediction often feeds the current CGM readings and other factors into the physiological models to predict the short-term blood glucose levels (*e.g.*, in 30 minutes) to allow for precautionary measures. One major drawback of physiological models is the requirement for prior knowledge to adjust the physiological parameters. To eliminate the need for acquiring physiological parameters, Plis *et al.* [44] apply a generic physiological model of blood glucose dynamics to extract features and adopt support vector regression to directly predict blood glucose levels. Reymann *et al.* [48] replace the physiological model by an online simulator to pre-calculate and validate the parameters for blood glucose prediction on mobile platforms.

In addition to the generic physiological models, various personalized external factors such as meals, insulin intake, exercises, and sleep quality, etc., can also lead to blood glucose changes [24]. METABO [17] is a client-server architecture based system that records dietary, physical activity, medication and medical information

for hypoglycemic and hyperglycemic event prediction. Marling *et al.* [35] improve hypoglycemia detection by combining CGM data with heart rate, galvanic skin response and skin temperature collected from a fitness band.

The feature engineering of SugarMate is built upon the general physiological models and the user-specific external factors. However, instead of feeding CGM measurements as input for *every prediction*, SugarMate explores to infer the *current* blood glucose level using *historical* blood glucose records and the *current* external factors. SugarMate offers the opportunity for users to *temporarily* take off the CGM devices and only rely on smartphones for blood glucose level monitoring.

2.2 Blood Glucose Monitoring with Alternative Sensors

As most CGM devices are inconvenient to wear for an extended duration of time and require complicated maintenance, there has been attempt at non-invasive blood glucose monitoring with pervasive wearable and mobile devices. Nguyen *et al.* [37] observe distinct patterns in ECG signals during hypoglycemia and hyperglycemia in Type I diabetic patients. Sobel *et al.* [50] integrate five types of sensory data from two accelerometers, a heat-flux sensor, a thermistor, two ECG electrodes and a galvanic skin response sensor to predict blood glucose concentration.

Our work is inspired by this body of research. SugarMate takes one step further by using smartphones rather than complex multi-sensory platforms [37] [50] and conducts evaluations with both diabetic and non-diabetic participants. In addition, SugarMate investigates the feasibility of blood glucose monitoring without CGM devices for a longer time (*e.g.*, re-calibration every three weeks), while previous works only show a possibility of 3 hours [50].

2.3 Machine Learning for Blood Glucose Monitoring

Despite extensive research on mobile sensing [31] [22] [26] [57], most works on blood glucose monitoring leverage simple machine learning techniques such as Support Vector Machines (SVMs) [44]. In SugarMate, we apply Recurrent Neural Networks (RNN) [42], which are effective for sequential inference. RNNs have been widely adopted in applications such as handwriting recognition [18] and speech recognition [19]. Another crucial factor in blood glucose monitoring is the need for personalized learning, so as to reflect user-specific factors, such as age, weight and insulin-to-carbohydrates ratio [41]. Nevertheless, a primary impediment to build up such models is the lack of sufficient personalized blood glucose data [35]. In this work, we propose Md³RNN, a multi-division RNN framework for blood glucose level monitoring. It shares blood glucose information among groups of users, but preserves user-specific blood glucose characteristics via personalized learning. Although many successful RNN architectures have utilized a shared feature extraction method (*e.g.*, auto encoding) as input to a personalized output [30], Md³RNN not only shares personalities, but also encodes grouped and embedded features into a share layer, which comprehensively captures the characteristics of different diabetic types.

3 OVERVIEW

SugarMate is a smartphone-based blood glucose level tracking system that (i) non-intrusively collects important external impacting factors and conducts feature engineering, (ii) efficiently trains a personalized blood glucose level model, and (iii) automatically provides reminders to users of abnormal blood glucose levels. Fig. 1 shows the architecture of SugarMate, which consists of three modules.

The **external factor collection and feature engineering** module records external factors that are important to infer blood glucose levels. A user records daily meal, drug and insulin intake. Meanwhile, SugarMate automatically measures physical activities and sleep quality via embedded sensors (*i.e.*, accelerometer, microphone and light sensor). After collecting data from multiple users, SugarMate conducts feature engineering from physiological and temporal perspectives, and feeds them into **Md³RNN**, a multi-division deep learning framework specific designed for blood glucose inference. Md³RNN first learns feature representations from users in the same

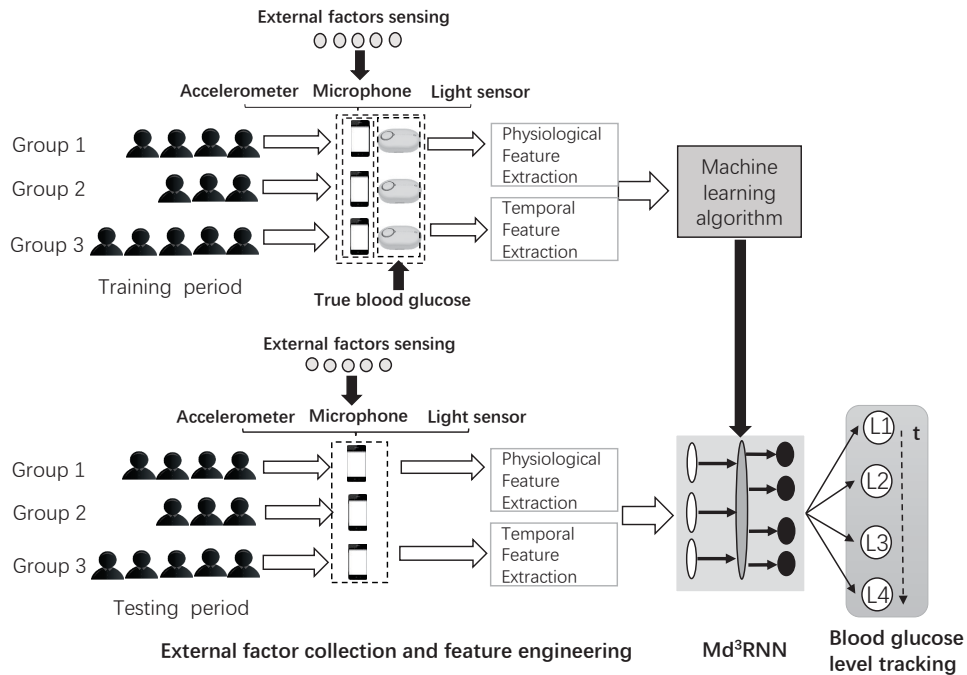


Fig. 1. Architecture of SugarMate.

group (non-diabetic, Type I and Type II diabetic), and then adopts a deep RNN layer to learn a general blood glucose level model on the dataset of all users. Finally, it outputs a personalized blood glucose level model for each individual via a personality layer. The inference results are eventually shown in the **blood glucose level tracking** module at a fine grained time resolution (every 3 minutes by default). In SugarMate, we consider 4 blood glucose levels as in Table 1.

Table 1. Normal and abnormal blood glucose levels [56].

Blood Glucose Value (mmol/L)	Glucose Level	Explanation
(0, 4.4]	Level 1	Low blood glucose (hypoglycemia)
(4.4, 6.1]	Level 2	Normal level of fasting blood glucose
(6.1, 7.8]	Level 3	Normal level of postprandial blood glucose
(7.8, +∞)	Level 4	High blood glucose (hyperglycemia)

4 DESIGN

This section presents the detailed design of SugarMate.

4.1 External Factor Sensing

It is neither possible nor necessary to exhaust a complete list of influential factors on blood glucose concentration. In SugarMate, five major external factors are measured, including food, drug and insulin intake, physical activities and sleep quality.

Food Intake. SugarMate provides a food menu for users to keep track of their meals. Based on the carbohydrate food list [40], meals are categorized into five types, including grains, vegetables, milk and egg, fruits, and meat. Users are asked to enter the food types and amounts of their meals, based on which SugarMate calculates U_C , the carbohydrate of a meal.

Drug Intake. Oral diabetic drugs enhance the secretion of insulin into the blood and are usually used by Type II diabetic patients. In SugarMate, a menu of 11 common oral medicines is presented for users based on [5]. Users are required to select the drug name and record the dosage.

Insulin Injection. Insulin injection is widely used for blood glucose control for Type I and Type II patients. SugarMate provides an insulin type list based on [4] for users to record the usage and dosage of their insulin injection. SugarMate automatically transforms drug intake and insulin injection into the amount of acting insulin U_I via bolus and basal rate information [44].

Physical Activity. Daily activities *e.g.*, exercises, consume the carbohydrate and affect blood glucose levels. In SugarMate, we adopt an efficient activity recognition scheme [29], which leverages the accelerometer to automatically record six common physical activities (walking, running, going upstairs, going downstairs, sitting and standing) along with the corresponding durations. SugarMate then calculates the caloric expenditure using a calorie calculator [36].

$$\text{CalorieBurn} = (\text{BMR}/24) \times \text{MET} \times T, \quad (1)$$

where BMR (Basal Metabolic Rate) is the amount of energy required to simply sit or lie quietly, and MET (Metabolic Equivalent) is the ratio of the work metabolic rate to the resting metabolic rate. T is the activity duration time (in hours). SugarMate leverages the caloric expenditure U_E as input for physiological feature extraction.

Sleep Quality. Sleep quality has a long-term influence on the blood glucose level [24]. SugarMate automatically measures sleep quality using [21], which invokes the accelerometer, microphone and light sensor for sleep quality inference. The output sleep quality score U_S is used for physiological feature extraction.

In summary, the five external factors are transformed into four categories of measurements including the carbohydrate of a meal (U_C), the amount of acting insulin (U_I), caloric expenditure (U_E) and sleep quality score (U_S), which are then used as inputs to extract important features for blood glucose level inference.

4.2 Feature Engineering

We extract features $X = \{X_P, X_T\}$ from external sensory data $U = \{U_C, U_I, U_E, U_S\}$ from both the physiological view (X_P , 10-dimension) and the temporal view (X_T , 51-dimension) to infer blood glucose levels.

4.2.1 Features from Physiological View X_P . Physiological features describe the dynamics of glucose related variables [9] [44]. We extract physiological features based on the physiological model in [9], which characterizes carbohydrate dynamics, insulin dynamics, exercise dynamics, and glucose dynamics. We further include sleep dynamics, another important external factor that affects blood glucose levels [24]. Specifically, the following physiological features are extracted to represent blood glucose relevant dynamics.

- Features from *carbohydrate dynamics*: carbohydrate consumption C_{g_1} and carbohydrate digestion C_{g_2} .
- Features from *insulin dynamics*: subcutaneous insulin absorption I_a , insulin secretion by pancreas I_s , insulin mass I_m and active plasma insulin level I .
- Features from *exercise dynamics*: long-term effect of exercises on insulin E .
- Features from *sleep dynamics*: long-term effect of sleep quality on insulin S .
- Features from *glucose dynamics*: glucose mass G_m and glucose concentration G .

The features are inter-dependent and are also related to the external factors. Fig. 2 illustrates the dependencies among variables and the corresponding transition equations to calculate the physiological features at time $k + 1$ using external factor measurements at time $k + 1$ and historical physiological features at time k . We briefly

summarize the transition relationships among these dynamics and refer interested readers to [9] for detailed derivation of the equations.

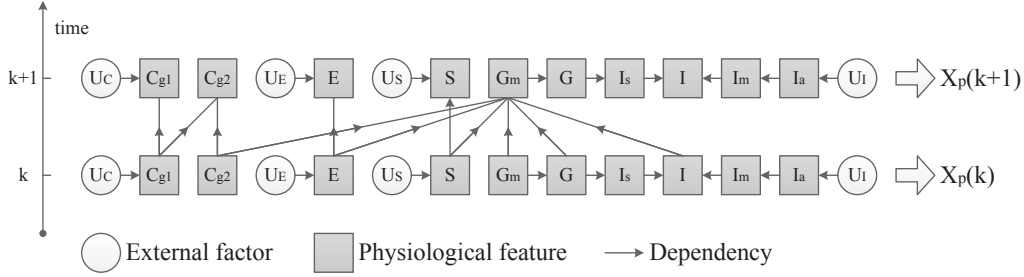


Fig. 2. An illustration of dependencies of physiological features $X_P = \{C_{g1}, C_{g2}, I_a, I_m, I_s, I, E, S, G_m, G\}$ and of external factors $U = \{U_C, U_I, U_E, U_S\}$.

Carbohydrate dynamics: Carbohydrate dynamics refer to the transitions of carbohydrate consumption C_{g1} and the carbohydrate digestion C_{g2} (see Eq.2 and Eq.3). U_C stands for the carbohydrate of meals.

$$C_{g1}(k+1) = C_{g1}(k) - \alpha_1^c \times C_{g1}(k) + U_C(k) \quad (2)$$

$$C_{g2}(k+1) = C_{g2}(k) + \alpha_1^c \times C_{g1}(k) - \alpha_2^c \times C_{g2}(k) \quad (3)$$

We choose carbohydrate consumption C_{g1} and carbohydrate digestion C_{g2} as part of the feature vector.

Insulin dynamics: Insulin dynamics indicates the transitions of subcutaneous insulin absorption I_a , the insulin secretion by pancreas I_s , the insulin mass I_m , and the level of active plasma insulin I (see Eq.4, Eq.5, Eq.6 and Eq.7). U_I is the amount of insulin injected or simulated by the diabetes drugs, and S^I and bm refer to the insulin sensitivity and body mass.

$$I_a(k+1) = I_a(k) - \alpha_{f,r,m}^I \times I_a(k) + U_I(k) \quad (4)$$

$$I_s(k+1) = \begin{cases} I_s(k) + \min[\alpha_1^I(\alpha_2^I(G_k - G_0)) + \alpha_3^I \times G_0, \Delta I_{max}^s] & \text{Type II} \\ I_s(k) + 0 & \text{Type I} \end{cases} \quad (5)$$

$$I_m(k+1) = I_m(k) + \alpha_{f,r,m}^I \times I_a(k) + \alpha_a^I \times I_a(k) - \alpha_c^I \times I_m(k) \quad (6)$$

$$I(k) = (I_m(k) \times S^I) / (142 \times bm) \quad (7)$$

We choose subcutaneous insulin absorption I_a , insulin secretion by pancreas I_s , insulin mass I_m and active plasma insulin level I as part of the feature vector.

Exercise dynamics: Exercise dynamics E denotes the exercise effect on insulin over the past time window. This long-term influence can be expressed by a cumulative moving average as Eq.8.

$$E(k - k_0 + 1) = (k - k_0) \times E(k - k_0) + U_E(k - k_0) \quad (8)$$

where k and k_0 are the current and beginning time points in the past time window. In SugarMate, the window size of exercise is set to 24 hours, which optimizes the experimental results. U_E denotes the calories cost of the exercise. We choose the long-term effect of exercises on insulin E as part of the feature vector.

Sleep dynamics: Sleep dynamics S represents the sleeping quality effect on insulin. Sleep has a constant effect on blood glucose for each day. Eq.9 shows its transition.

$$S(k - k_0 + 1) = (k - k_0) \times S(k - k_0) + U_S(k - k_0) \quad (9)$$

where k and k_0 are the current and beginning time points in the past time window. In SugarMate, the window size of sleep lasts for 7 days, which optimizes the experimental results and matches the conclusion of clinical studies. U_S is the sleep quality score. We choose the long-term effect of sleep quality on insulin S as part of the feature vector.

Glucose dynamics: Blood glucose dynamic refers to the fluctuation of glucose mass G_m in Eq.10 and glucose concentration G in Eq.11.

$$G_m(k+1) = G_m(k) + \delta_{abs} + \delta_{egp} - \delta_{ind} - \delta_{dep} - \delta_{clr} \quad (10)$$

$$G(k) = G_m(k)/(2.2 \times bm) \quad (11)$$

δ_{abs} is the impact of carbohydrate absorption and δ_{egp} is the hepatic glucose production from the liver. These two factors increase the blood glucose level and their calculation methods are listed in Eq.12 and Eq.13

$$\delta_{abs} = \alpha_3^c \times \alpha_2^c \times C_{g2} \quad (12)$$

$$\delta_{egp} = \alpha_2^{egp} \times \exp(-I(k)/\alpha_3^{egp}) - \alpha_1^{egp} \times G(k) \quad (13)$$

δ_{ind} describes the result of insulin independent uptake, which is consumed by the central nervous system and the red blood cells. δ_{dep} indicates the impact of insulin dependent uptake. It reflects the effects of insulin promoting muscle cells and fat cells to absorb glucose, including the influence of sleeping and exercise factors. δ_{clr} stands for the influence of renal clearance on blood glucose. Once the blood glucose concentration exceeds the renal clearance threshold τ , the kidneys begin to remove excess glucose from the blood. These three factors decrease the blood glucose level, and their calculation methods are listed in Eq.14, Eq.15 and Eq.16.

$$\delta_{ind} = \alpha_1^{ind} / \sqrt{G(k)} \quad (14)$$

$$\delta_{dep} = \alpha_1^{dep} \times E(k) \times S(k) \times I(k) / (G(k) + \alpha_2^{dep}) \quad (15)$$

$$\delta_{clr} = \alpha_1^{clr} \times (G(k) - \tau) \quad (16)$$

Note that the above transition equations involve a set of parameters α 's, e.g., the insulin sensitivity S^I and the body mass bm . These parameters are user-specific parameters that need to be tuned per-person. The default values of parameters in the physiological model are set based on [9] and are further tuned for each person via 10-cross validations.

In summary, we extract a 10-dimension physiological feature vector X_P .

4.2.2 Features from Temporal View X_T . As blood glucose level naturally varies over time, we extract two temporal features for blood glucose level inference.

- Historical blood glucose concentration X_{T_1} at time k : Since people usually tend to lead a regular lifestyle, e.g., having meals in the morning, at noon and in the evening and taking drugs at certain times, the blood glucose concentration also exhibits rough daily cycles. Fig. 3 plots the daily blood glucose traces of a volunteer for five successive days measured by a CGM device. As shown, the blood glucose traces always grow up significantly in the durations of 8:40 to 9:40, 12:30 to 14:00 due to the breakfast and lunch, but increase moderately from 18:00 to 20:00 because of taking the drugs before the dinner. This motivates us to adopt the historical blood glucose concentrations at time stamp k (averaged over D days) as one temporal feature to infer the blood glucose level at time k . In SugarMate, we set $D = 5$ and infer blood glucose level at a time resolution of 3 minutes, which is in accordance with the time resolution of commercial CGM devices [34].

- Most recent physiological features X_{T_2} : As shown in the physiological models, the current blood glucose concentration is relevant to the recent blood glucose concentration and physiological features. However, in the physiological features X_P , only the physiological features at the last time stamp are considered. To account for more short-term temporal dependencies, we propose to include the l most recent physiological features $X_{T_2}(k) = \{X_P(k), X_P(k-1), \dots, X_P(k-l+1)\}$, where $l = 5$ in our implementation. That is, instead of considering the physiological features in the last 3 minutes, we infer the current blood glucose level by leveraging features in the last 15 minutes.

In summary, we extract a 51-dimension $(1 + 10 \times 5)$ temporal feature vector X_T .

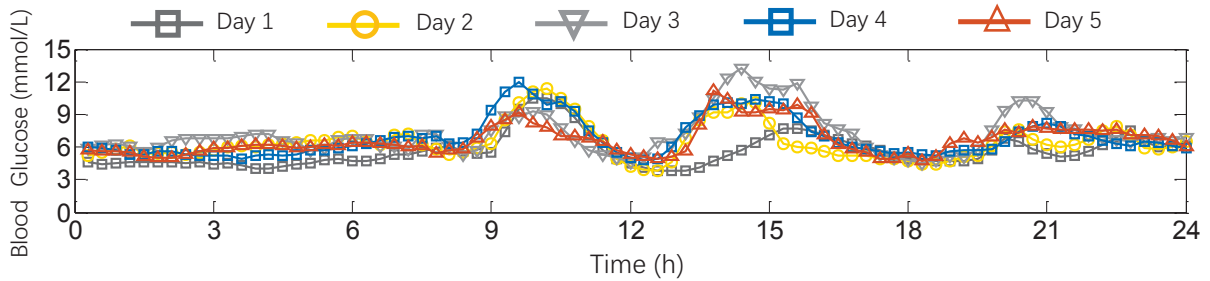


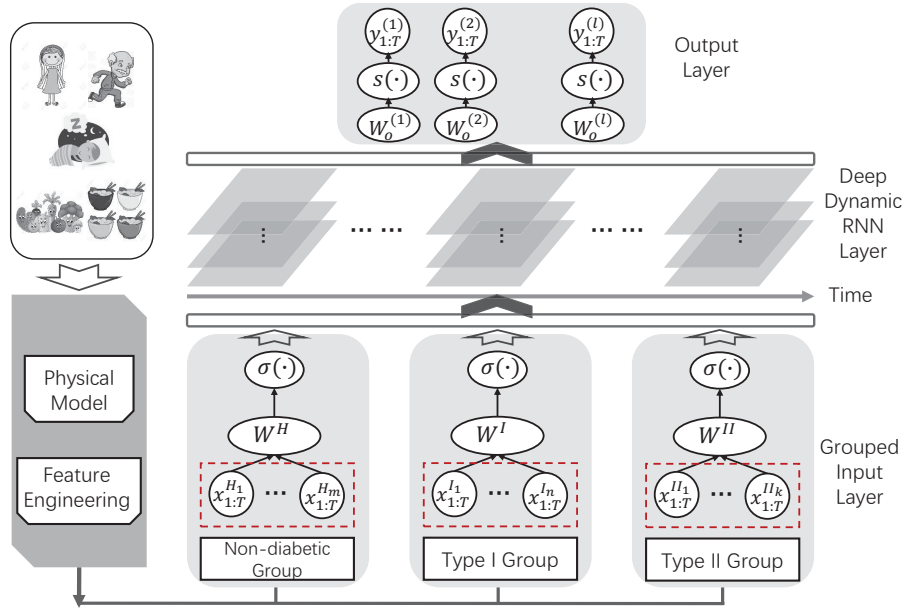
Fig. 3. Daily blood glucose traces of a volunteer. We only plot 10 measurements every 3 hours for ease of illustration.

4.3 Blood Glucose Level Inference

Given the features extracted based on the physiological process, it seems plausible to perform any classification algorithm for blood glucose level inference. Nonetheless, this plug-and-play approach will neglect important information from (1) dynamics of the process, and (2) inter-user similarity among the same group of participants. Traditionally, various sequential methods, *e.g.*, hidden Markov models (HMM), recurrent neural networks (RNN) and time series models, are used to capture the temporal correlation of the input feature. The inter process correlations are often times incorporated with a co-regularized approach like [12], which learns processes in parallel to improve classification or to reduce the data sample requirement.

In this paper, a novel machine learning paradigm, namely Multi-division deep-dynamic RNN (Md^3RNN), is proposed. To include the the aforementioned information sources in an unified framework, we develop two key ideas that extend the classical RNN. Firstly, the single hidden layer in RNN is replaced with several deep stacked layers. The deep structure in the new model is able to describe complex, multi-scale dynamics that would otherwise be ignored (or averaged out) by prior “shallow” models. Secondly, the correlations among users, being quite significant within user groups (divisions), are encoded by group-shared input layer and common hidden layers, whereas the distinct characteristics of individual users are modeled with different output layers for personalized prediction. Within a larger scope of machine learning, the proposed Md^3RNN aims to leverage recent advancements of deep learning and structured representation learning [59], to model group-interacted time series data having complex temporal dynamics. It can be viewed as both a deep extension of RNN, and an improved version of generic learning method for data with a multi-division structure, hence the name. Although we develop Md^3RNN for the specific use case of SugarMate, it is worth pointing out that it can be readily applied to many other applications dealing with grouped dynamic data. The overall configuration of the proposed model is summarized in Fig. 4. Detailed construction of each component is given in the sequel.

4.3.1 Model construction by layers. The inputs of the Md^3RNN model are the features extracted following the discussion in the previous section. The labeled data sequences for user number j at time i are denoted by (x_i^j, y_i^j) .

Fig. 4. Illustration of Md³RNN structure.

We also adopt an index set convention, that (x_A^B, y_A^B) represents the data set $\{(x_i^j, y_i^j) | i \in A, j \in B\}$ given index sets A and B .

Grouped Input Layer. In the context of blood glucose prediction, available inputs are naturally divided into three groups according to the health condition of the participant from whom the data was generated. Notation-wise, we utilize H, I and II to indicate the the group of non-diabetic user (healthy), user with type I diabetes and those having type II diabetes, respectively. Since the extracted features are essentially physiological indexes and temporally correlated variables, they must go though different transformations to represent useful information of three distinct groups. This consideration motivates the design of the input layer (bottom of Fig. 4) - it is divided into three units that performs different linear and non-linear transformations according to user groups. For instance, a data sample x_t^{Ij} , generated at time t from the j^{th} user of type I, undergoes the following processing:

$$\tilde{x}_t^{Ij} = \sigma(W^I x_t^{Ij}) \quad (17)$$

where W^I is the coefficients of the affine transformation², σ is the sigmoid activation function, and \tilde{x}_t^{Ij} is the output of the input layer for that data sample. Similar operations are conducted for data samples from group H and II , but with different transformation coefficients. Intuitively, the shared transformation within groups would improve the learning of parameters (vs. single task learning), as information from all data in a homogeneous group is used. Also, the transformation can be stacked into several (say P) layers, for better information representation.

Deep Dynamic Layer. A common hidden layer is designated to capture the dynamics of the blood glucose evolution process. The underlying assumption is that, the physiological reactions governing blood glucose variation are similar for all people, despite grouped behaviors in the representation of physiological indexes (input layer), or individual characteristics in exhibited glucose level. This assumption could be justified by a series

²We assume that the interception is included in W . This can be done by simply adding a constant feature vector of 1.

of medical related research [32] [8] [23] [9]. Moreover, since all users share the same hidden layer, all collected data samples are eventually helping the estimation of its parameters. The availability of rich information for the hidden layer makes the learning of a deep structure possible. In SugarMate, a number of Long Short Term Memory (LSTM) networks are stacked together (middle of Fig. 4), to increase the overall model flexibility. It has been justified in both theory and practice that stacked LSTMs are able to capture dynamics occurring at different time scales [10], which in the current application would enable the modeling of both slow and rapid biological/chemical reactions. Although a wide variety of LSTM configurations exist in literature, in this work we adopt the one recently proposed by [28], which combines the forget/input gate and merges cell/hidden state for simplicity and better generalization performance. Mathematically, given the output from the grouped input layer, the deep dynamic layer performs

$$\begin{aligned}
z_t^d &= \sigma^d \left(W_z^d [h_{t-1}^d, h_t^{d-1}] \right) \\
r_t^d &= \sigma^d \left(W_r^d [h_{t-1}^d, h_t^{d-1}] \right) \\
\tilde{h}_t^d &= \tanh \left(W_h^d [r_t^d * h_{t-1}^d, h_t^{d-1}] \right) \\
h_t^d &= (1 - z_t^d) * h_{t-1}^d + z_t^d * \tilde{h}_t^d
\end{aligned} \tag{18}$$

for hidden layer numbered $d = 1, 2, \dots, D$. At the first dynamic layer with $d = 1$, the input h_t^{d-1} is set to be the output from the grouped input layer, and the output of the last dynamic layer, h_t^D , will be used as the input of the last component of Md³RNN.

Personalized Output Layer. Finally, each user is assigned a personalized output layer, parameterized by $W_o^j, j = 1, \dots, l$, which performs a single linear and softmax transformation on the results of the deep dynamic layer. The particular configuration of the output layer compensates for the individual characteristics in the exhibited blood glucose (*i.e.*, measured blood glucose level). Because only data generated by a specific participant j will have an effect on the parameters W_o^j , the personalized output layer is set to have a “shallow structure”, *i.e.*, it only performs the transformation once. More specifically, given h_t^D from user j , it computes

$$\hat{y}_t^j = \text{softmax} \left(W_o^j h_t^D \right) \tag{19}$$

4.3.2 Cost Sensitive Learning and Hyperparameter (Model) Selection. Similar to other deep neural network learning, Md³RNN is trained by minimizing the sum of losses over all the time steps. The definition of the loss function has much bearing on the generalization performance of the method [58]. In particular, for the current application, simply minimizing a general error rate seems inappropriate, because the costs of different types of misclassification errors can differ a lot. For example, missing the detection of high blood glucose (type I or II) is more costly than misclassifying normal condition to an alarm for high glucose. Moreover, in the collected data set from real people, the training data is inherently imbalanced - the available samples labeled Level 1 and Level 4 are much fewer (only 30.6% in training dataset) compared to samples in the other categories (Level 2 and Level 3).

The above concerns motivate the cost sensitive learning of Md³RNN. Instead of directly minimizing a surrogate of error rate, we propose to optimize over a weighted version of classification losses. More specifically, the following total loss function L is considered:

$$\begin{aligned}
L &= \sum_{j \in H} \sum_{t=1}^T \sum_{y_t \in \mathcal{Y}} l^H(y_t, \hat{y}_t) C_{y_t} + \sum_{j \in I} \sum_{t=1}^T \sum_{y_t \in \mathcal{Y}} l^I(y_t, \hat{y}_t) C_{y_t} + \sum_{j \in II} \sum_{t=1}^T \sum_{y_t \in \mathcal{Y}} l^{II}(y_t, \hat{y}_t) C_{y_t} \\
&\quad + \lambda^H \|W^H\|_{2,1} + \lambda^I \|W^I\|_{2,1} + \lambda^{II} \|W^{II}\|_{2,1} + \lambda^d (\|W_z\|_F + \|W_r\|_F + \|W_h\|_F) + \lambda^o \|W_o\|_F
\end{aligned} \tag{20}$$

where \mathcal{Y} is the label set, and \hat{y}_t^j s are prediction outputs at time stamp t . Our implementation uses cross entropy as individual loss function $l(y_t^j, \hat{y}_t^j)$, but generally the “base” loss can be any surrogate of the error function. The additional coefficient C_y weights the misclassification error for category y . In the current application, $\mathcal{Y} = \{1, 2, 3, 4\}$, associated with four coefficients C_1 to C_4 . Those cost weighting coefficients are treated as hyperparameters of the proposed model, but in other applications of Md³RNN, they can also be determined with prior knowledge about the misclassification cost and the class imbalance. The other terms in Eq.20 are regularizations for model parameters, which in the current implementation is substantiated with the $L_{2,1}$ norm on the parameters of the input layer for better information extraction [39], and the frobenius norm on the parameters of the dynamic and output layer to penalize model complexity. The coefficients λ s are left as model hyper-parameters, whose selection procedure will be discussed later.

With the technique of back-propagation, computing the gradient of Md³RNN is not so different from the gradient calculation of classical RNN. In this work, we accomplish those computation using Tensorflow [1], and proceed to learn Md³RNN model by stochastic gradient descent for overall loss minimization.

Last but not least, the construction of the Md³RNN model involves choosing 15 hyperparameters, *e.g.*, cost coefficients C_y , depth D of the stacked dynamic layer, learning rate, number of hidden unit in the input layer, etc. Direct application of cross validation (CV) for hyperparameter tuning, even with the help of parallel computing, seems intractable as the number of required CVs scales exponentially to the number of hyperparameters. In this regards, we adopt Bayesian optimization (BO), a recent tool developed for blackbox function optimization with limited evaluations. The decision variables of BO are those hyperparameters, and the objective is the F-score of the precision and recall on some testing data set. Note that BO has been used recently for the hyperparameter (model) selection of many deep learning paradigms [49].

5 EVALUATION

5.1 Experimental Settings

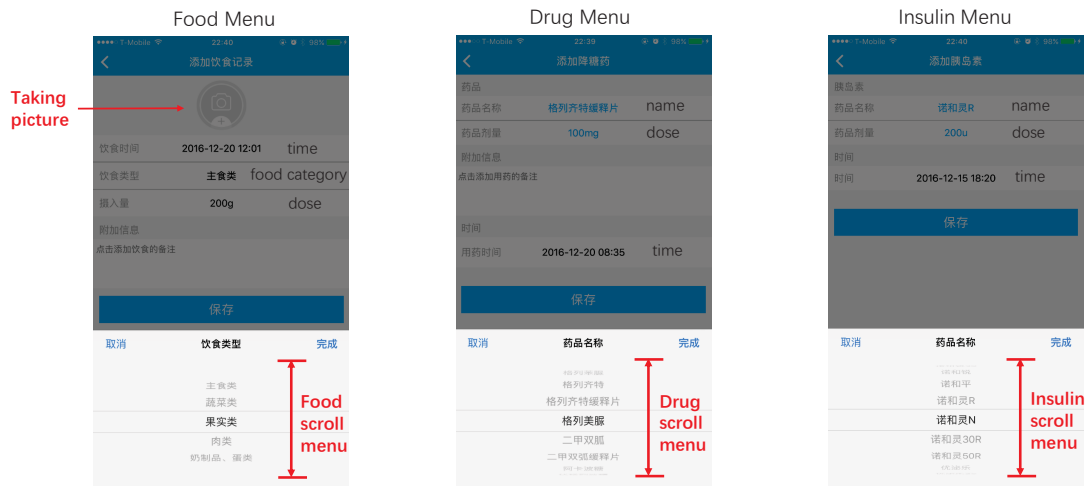


Fig. 5. User interfaces for food, drug and insulin intake recording.

Datasets. We validate SugarMate on a dataset of 112 participants (35 non-diabetes, 38 type I diabetic patients and 39 type II diabetic patients) collected from July 2016 to January 2017. Each participant is equipped with

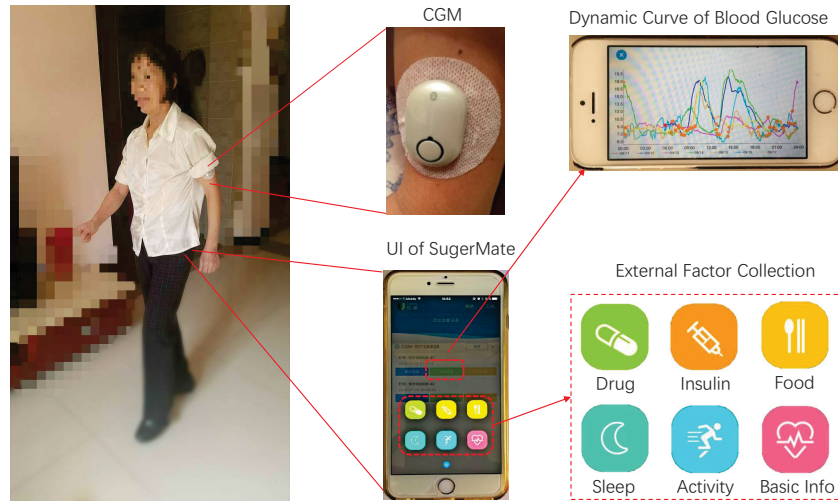


Fig. 6. An illustration of the equipments for data collection. Each participant wears a CGM device to record blood glucose concentration and uses a smartphone to collect external factors.

(1) a WAVEGUIDER *U-Tang* CGM device [34] to record blood glucose concentration every 3 minutes and (2) a smartphone with SugarMate installed to collect external factors either automatically (activities and sleep quality) or manually (food, drug, and insulin intake). Fig. 5 shows the user interfaces for manual recording of food, drug and insulin intake. All participants agree to take measurements (*i.e.*, wear the CGM device and use SugarMate to record external factors) at least 6 days, which is a normal disposable usage duration of the enzyme in the sensor of the CGM. Fig. 6 illustrates an example of data collection from a user. We record measurements for each participant from 6 to 30 days. In total we obtain 762639 samples of blood glucose concentration and the corresponding external factors covering around 38132 hours. In brief, we collect the following categories of data:

- **Meta information.** We record basic personal data including gender, age, weight and health status to cover a wide range of users. Table 2 summarizes the basic information of the participants.
- **Blood glucose measurements.** We collect blood glucose measurements using commercial CGM devices for 6 to 30 days as labeled data. Table 3 summarizes the blood glucose measurements in our evaluation.
- **External factor measurements.** During measurements of blood glucose concentration, each participant manually inputs the times of their daily meal, drug and insulin intake. SugarMate automatically records activity levels and sleep quality as in Sec. 4.1. Fig. 6 shows the user interfaces to record external factors.

Ground Truth. We use the blood glucose concentrations collected by the CGM device as ground truth³.

Metrics. We mainly adopt precision, recall and accuracy to quantify the performance of SugarMate.

5.2 Inference Accuracy

5.2.1 Overall Inference Accuracy. Since all participants collected both measurements of CGM and external factors for at least 6 days, we use measurements during the former 5 days for training and the rest for testing. Table 4 shows the overall performance of SugarMate. All results are averaged over the testing data. As shown, the recalls and the precisions for all the 4 blood glucose levels are above 79% and 73%, respectively. In particular, the

³While clinical studies report that the precision and accuracy of commercial CGM devices still need improving [51], they are sufficient as ground truth for the four normal and abnormal blood glucose levels.

Table 2. Summary of participant information.

(a)		(b)			(c)	
Age (year)	# User	Weight	BMI (kg/m^2)	# User	Gender	# User
15-24	8	Underweight	(0, 18.5)	18	Male	57
25-34	17	Normal weight	[18.5, 25)	31	Female	55
35-44	24	Overweight	[25, 30)	41		
45-54	29	Obese	[30, $+\infty$)	22		
55-64	34					

Table 3. Summary of blood glucose measurements.

(a)		(b)	
Duration (days)	# User	Blood Glucose	# Sample
6-10	48	Level 1	75369
11-15	24	Level 2	293530
16-20	20	Level 3	235686
21-25	13	Level 4	158054
26-30	7	Total	762639

recalls for Level 1 (low blood glucose) and Level 4 (high blood glucose) are 83.13% and 85.23%, even though the training data for Level 1 and Level 4 only take up around 10% and 20% of the entire training set. This result shows that SugarMate can accurately infer low/high blood levels even with an imbalanced training dataset. Overall, SugarMate yields an accuracy of 82.14%, showing a promising performance to track blood glucose levels. Note that SugarMate is not designed to substitute CGM devices or medical measurements (e.g., direct finger sticker). However, the precisions and recalls in low and high blood glucose level inference make SugarMate suitable to remind users of abnormal blood glucose levels so that they can double-check by finger stickers and take the corresponding treatments.

Table 4. Confusion matrix of SugarMate.

Ground Truth	Inference				Recall
	Level 1	Level 2	Level 3	Level 4	
Level 1	62657	5521	3672	3519	83.13%
Level 2	16346	240584	27563	9037	81.96%
Level 3	2660	30905	188472	13649	79.97%
Level 4	3443	5620	14278	134713	85.23%
	73.62%	85.12%	80.55%	83.72%	Accuracy: 82.14%
	Precision				

5.2.2 Inference Result Analysis. To understand the inference accuracy and the risks of different types of errors in the context of blood glucose management, we classify the inference results based on the principles of Clarke

Error Grid Analysis (CEGA) [7]. The analysis classifies the inference results into correct event (Type A) and different types of errors (Type B to Type E) with increasing levels of severity. For instance, Type B errors are those that will not lead to inappropriate treatments, while Type E errors can lead to wrong treatment. Table 5 summarizes the percentages of each type of results. As shown, SugarMate will not cause inappropriate treatment (Type A and B) in almost 90% of the cases. It may lead to unnecessary worries or treatment (Type C) in 5.47% of the cases. In fewer than 5% of the cases, SugarMate will miss an abnormal blood glucose event (Type D) or confuse treatment (Type E). Therefore, SugarMate is suitable as an temporary alternative for CGM devices. However, we do not recommend SugarMate to those serious diabetic patients, who need professional clinic blood glucose managements.

Table 5. Clarke error grid analysis

Type of Result	Explanation of Result	Percentage
Type A	The inference value is consistent with the true value. (i.e., the inference blood glucose level is correct.)	82.14%
Type B	The inference result would not lead to inappropriate treatment. (i.e., Level 2 is predicted as Level 3, or vice-versa.)	7.67%
Type C	The inference result will lead to unnecessary treatment. (i.e., Level 2 is predicted as Level 1/4, or Level 3 is predicted as Level 1/4.)	5.47%
Type D	Fail to detect hypoglycemia or hyperglycemia. (i.e., Level 1/4 are predicted as Level 2/3.)	3.81%
Type E	The predicted results that would confuse treatment by mistaking hypoglycemia for hyperglycemia or vice-versa. (i.e., Level 1 is predicted as Level 4, and vice-versa.)	0.91%

5.2.3 Temporal View of Inference Results. Fig. 7 plots the inference results of SugarMate of three participants (one non-diabetic, one Type I diabetic patient, and one Type II diabetic patient) throughout a day. The errors are depicted at the bottom of each figure. As shown, the true blood glucose levels vary during the day after important daily activities such as food intake (5:50, 11:20 and 19:00 for the non-diabetic user; 6:00 and 16:50 for the Type I diabetic user; 6:00, 12:50 and 17:45 for the Type II user), insulin injection (7:40 for the Type I diabetic user), drug intake (15:10 for the Type II user) and exercises (15:30 for the Type II user), indicating the importance of external factors. The blood glucose levels inferred by SugarMate match the true blood glucose levels most of the time. Most errors mistake Level 2 and Level 3, and the errors often occur during the transition of two levels (e.g., from 6:10 to 6:30 for the Type II diabetic user), or in case of sudden change of blood glucose concentration (e.g., at 2:30 for the non-diabetic user and at 0:30 for the Type I diabetic user). Nevertheless, these errors belong to the Type B errors in Sec. 5.2.2, which will not lead to inappropriate treatment.

Table 6 shows the average false positives (FP) and false negatives (FN) per day and per hour for each user. Given an inference every 3 minutes or 480 inferences per day, the number of FPs and FNs per hour is no greater than two. Since only FPs for Level 1 and 4 will cause annoying notifications, such situations occur at an even lower rate. To further reduce the unnecessary notifications, SugarMate only reminds the user when there are three consecutive inferences of Level 1 or Level 4. This mechanism is acceptable because (a) most errors occur during transition of blood glucose levels or when there is a sudden change of blood glucose concentration, and (b) the 9-minute delay usually will still save sufficient time for proper treatment [16] [54].

5.3 Model Comparison

5.3.1 Effectiveness of Md^3 RNN Framework. To demonstrate the effectiveness of the multi-division framework in making full use of the dataset, we evaluate Md^3 RNN by 10-fold cross validation from two perspectives.

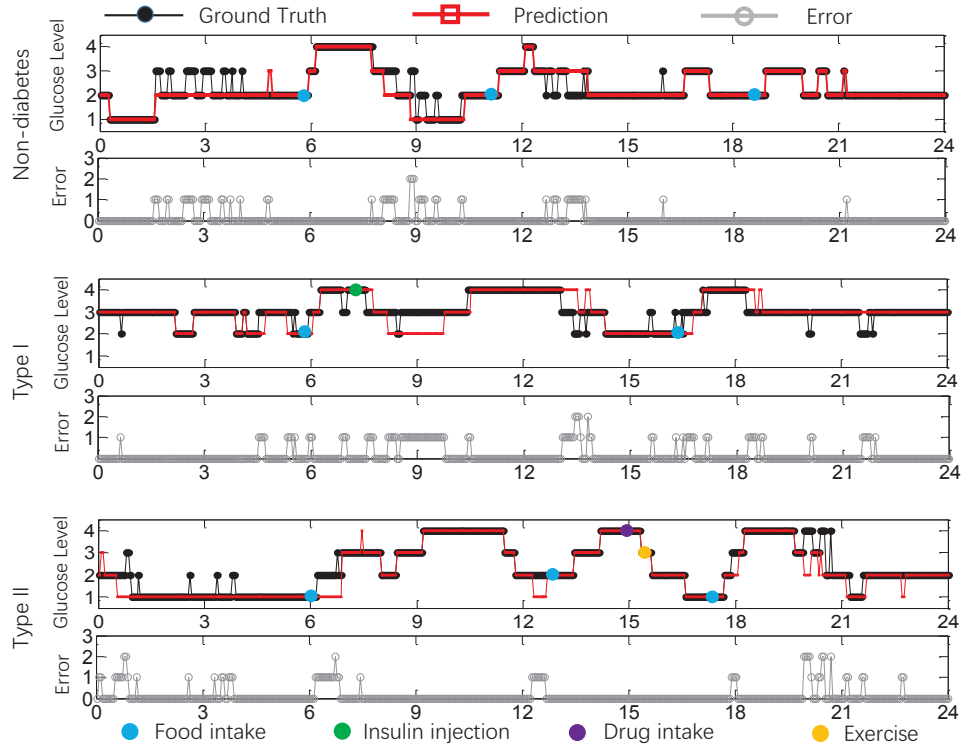


Fig. 7. Traces of blood glucose level inference results throughout a day.

Table 6. Summary of average false positives and false negative per user.

Level	FP (Day)	FP (Hour)	FPR	FN (Day)	FN (Hour)	FNR
Level 1	14.12	0.59	2.94%	8.00	0.33	1.67%
Level 2	26.46	1.10	5.51%	33.32	1.39	6.94%
Level 3	28.64	1.19	5.97%	29.71	1.24	6.19%
Level 4	16.49	0.69	3.44%	14.69	0.61	3.06%

Layer contribution analysis. To evaluate the effect of different layers, we conduct blood glucose level inference with three combinations of layers.

- *Deep dynamic layer.* Training without considering differences in groups and persons, and only output a general model.
- *Deep dynamic layer + Grouped input layer.* Learn group-specific feature representations but ignore per-person characteristics in the output.
- *Deep dynamic layer + Grouped input layer + personalized output layer ($Md^{\beta}RNN$).* Efficiently learn features from different groups and output personalized inference results.

Fig. 8 plots the comparison results of the three combinations. As shown, both the precisions and recalls increase with more layers, with an improvement of 21.13% in average precision and 18.57% in average recall, respectively. Moreover, the standard deviations (error bars) drop remarkably from 17.25% to 10.25% of average precision, and

from 20.75% to 10.75% of average recall. The results demonstrate the effectiveness of Md³RNN, which learns representative features from the same groups and considers individual differences in blood glucose level inference.

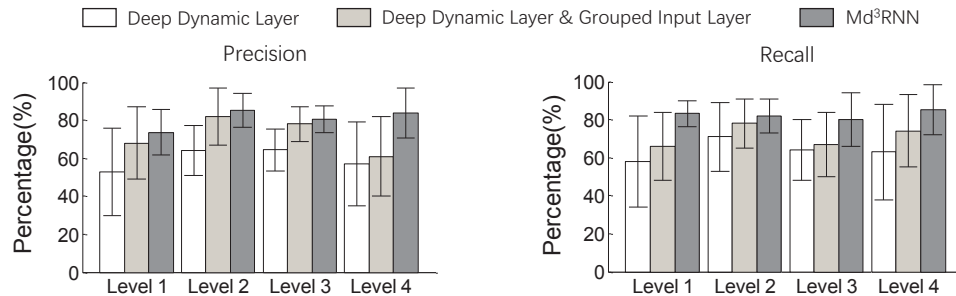


Fig. 8. Performance of layer combinations. The error bars denote the standard deviations on 10-fold cross-validation.

Comparison of data sharing schemes. To demonstrate the benefits of sharing data and knowledge among groups and users, we compare Md³RNN with other learning frameworks with different data sharing schemes.

- *General Learning.* All the training data are directly fed into the model (*i.e.*, deep RNN) for training indifferently. General learning results in a *generic* model that assumes universal correlations between all inputs and the blood glucose levels.
- *Group Learning.* The data of users belonging to a same group are fed into a model (*i.e.*, deep RNN) for training. Three separate models are obtained for three groups (*i.e.*, non-diabetic, type I and type II diabetic). The group learning results in a *group* model that shares the general characteristics of users within the same group but without data sharing among users in different groups.
- *Single Learning.* We train a different model (*i.e.*, deep RNN) for each individual participant by feeding his/her own measurements into the model. Single learning results in a *personalized* model without sharing data and learning knowledge from measurements of other participants.

Fig. 9 shows the overall precisions and recalls of our Md³RNN as well as *General learning*, *Group learning* and *Single learning*. As shown, our multi-divisional learning framework (Md³RNN) performs best among the four learning approaches with an average precision of 80.75% and an average recall of 82.57%. It also yields the lowest standard deviations (17.18% of average precision and 17% of average recall). The results show that Md³RNN is both effective and stable in blood glucose level inference.

General learning treats each sample of training data equally, and ignores the individual differences, so it performs poorly in most cases. Conversely, single learning encodes the individual characteristics but suffers from the lack of user-specific training data. It may require a very large personalized training set to achieve satisfactory performance. Even though group learning learns the similarities of users within the same group, it ignores inter-person physiological differences. Md³RNN combines the advantages of these three learning approaches, which makes better use of the limited individual training data by sharing measurements among users and preserves user-specific characteristics via the personal learning layer.

5.3.2 Effectiveness of Md³RNN Learning Algorithm. To demonstrate the effectiveness of the Md³RNN learning algorithm, we compare it with several typical algorithms from following two aspects.

Typical classifiers that do not share features from users. This group contains classical machine learning methods that are commonly used for standard learning applications. Note that in these traditional frameworks, the learning of each user's model is treated independently and the transferable similarities among users are

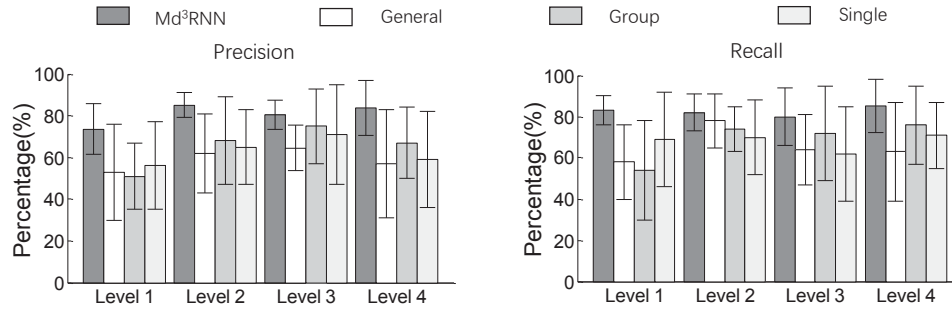


Fig. 9. Performance of data sharing schemes. The error bars denote the standard deviations on 10-fold cross-validation.

simply ignored. The following baselines, each with a very different modeling assumption, are included to justify the effort of information sharing proposed in our method.

- **Support Vector Machines (SVMs)** [52]. Support vector machines (SVM)s are supervised learning models with efficient convex learning algorithms that are widely used for classification and regression analysis. The idea is to construct optimal separating hyperplane that maximizes the separation margin of two data groups (classes). Due to this geometric property, it usually generalizes well, and its dual form is a quadratic programming that can be easily incorporated with kernels, which allows an implicit transformation of the examples from the original space to a non-linear high dimensional Hilbert space for better separation. We adopt the implementation of SVM in [43] for classification.
- **Gaussian Processes (GP)** [47]. Instead of directly parameterizing a latent function for classification, GP [47] models it with a generic Gaussian process, i.e., a distribution over the functional space of the classifier or regressor. The posterior of the process is updated with training data set, and is “squashed” through a logistic function for classification. The covariance matrix used in GP also allows the utilization of the “kernel trick” to capture similarities in some nonlinear space. The Gaussian process classifier utilized in this paper is provided by [43].
- **Hidden Markov model (HMM)** [46]. A hidden Markov model (HMM) is a statistical Markov model in which the system being modeled is assumed to be a Markov process with unobserved (hidden) states. It can be presented as the simplest dynamic Bayesian network. Simple as it is, HMM has been widely used in signal processing and time series analysis due to its interpretability and tractability.
- **Random Forest (RF)** [33]. As an ensemble method, RF combines many simple decision trees together and output the mode of classes for prediction. To avoid the correlation among base trees, random set of features are selected in the splitting process when constructing each decision tree.
- **Gradient Boosting (GB)** [14]. GB generates a prediction model by combining many weak classifiers into a stronger classification committee. We use the implementation of the fastAdaboost [25] to combine basic tree classifiers for ensemble learning.

Typical classifiers that allow sharing features among users. To evaluate the effectiveness of sharing features of Md³RNN, we compared it with several common and advanced machine learning frameworks, known as transfer or multi-task learning methods, that allow information sharing among multiple data sources or learning tasks. In the current scenario, a “learning task” refers to the blood glucose modeling of a particular user. The learning methods in this group attempt to improve the classification performance by incorporating similarities among users’ models. Specially, we consider the following:

- **Co-regularized Support Vector Machine (mSVM).** As a modified version of the classical SVM classifier, the authors of [12] proposed to incorporate the relation among tasks through a task-coupled kernel function. This consideration was then translated into a large margin learning framework with an additional co-regularization. Fortunately, the learning problem is still convex and the dual form still allows the the usage of kernels. We include this method also because its idea of co-regularization is the foundation of many other transfer learning or muti-task learning methods.
- **Hierarchical Gaussian Processes (mGP).** The information sharing of GP can be achieved by augmenting the kernel matrix to include side information among similar learning procedures. Since the augmentation step amounts to creating another layer of “similarity metric”, this approach is referred to as the hierarchical Gaussian Processes. In this exmperimtn, we adopt the version proposed in [2], and also implement several approximation algorithms for acceleration [6].
- **Nested Hidden Markov model (nDP-iHMM).** Classical HMM focus on the modeling of a signal random process. To further improve the flexibility of the model, the authors of [38] proposed the so-called Nested Dirichlet Process infinite Hidden Markov model (nDP-iHMM) based on a non-parametric method for possibly undetermined state space, and imposing a nested Dirichlet process prior to share information among tasks.
- **Hierarchical artificial neural network (hANN).** We also implement a hierarchical ANN based on classic ANN [53] as a baseline, simply to justify the benefit of “temporal structure engineering” of RNNs in Md³RNN. The ANN under comparison also contains "three divisions": a grouped input layer, three stacked time-independent hidden layers, and an output personal layer. The training of the hierarchical ANN is done by using the stochastic gradient descend algorithm implemented in Tensorflow [1].

Fig. 10 illustrates the results. Apparently, Md³RNN achieves best performance on both precisions and recalls. More specifically, the models that share features yield better performance than their corresponding models that do not share features (*i.e.*, mSVM vs. SVMs; mGP vs. GP; nDP-iHMM vs. HMM), which demonstrates the effectiveness of the methods that allow information sharing among tasks. Among those baselines, it appears that no method could dominate the others, except that hANN performs slightly better in terms of recall score. However, compared to Md³RNN, which is able to describe temporal dynamics by RNN, hANN is still worse in general. The dominating performance of Md³RNN is somewhat expected, as those baselines either ignore the multi-scale dynamics of the observed data, or can not allow information sharing among available data from users. The above observation further justifies the efforts of designing the multi-division deep dynamic RNN for SugarMate, which efficiently transfers valuable knowledge between groups and individuals.

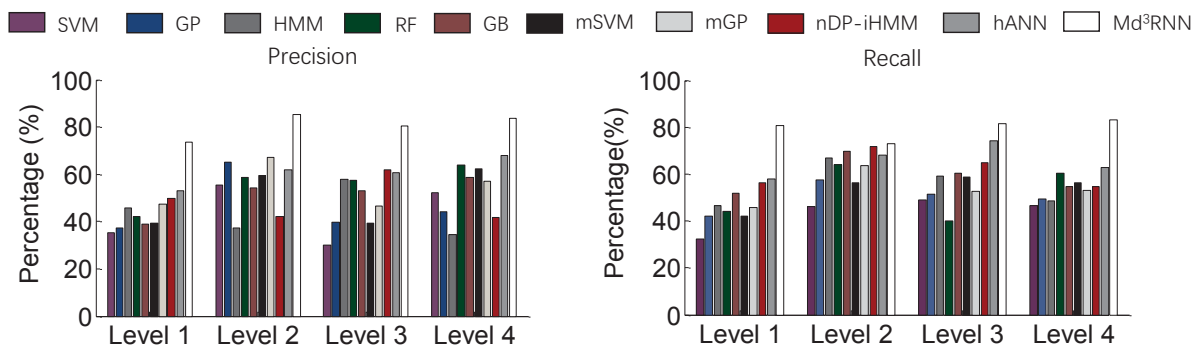


Fig. 10. Performance comparison with the different learning algorithms.

5.4 Micro-benchmarks

5.4.1 Effectiveness of Features. Table 7 shows the average precisions and recalls using different combinations of features. By combining physiological features (X_P) with temporal features (X_T), the average precision and recall of the 4 blood glucose levels improve by 31.38% and 41.48% respectively. The precision and recall increase by about 10% to 30% with physiological and short-term temporal features ($X_P + X_{T_2}$) over physiological features alone. However, the historical trends X_{T_1} prove to be more effective than the short-term temporal features X_{T_2} (see the second and the third rows of Table 7). The necessity to include historical trends indicates the need for re-calibration, as will be discussed in Sec. 5.4.3.

Table 7. Effectiveness of features.

Features	Level 1		Level 2		Level 3		Level 4	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
X_P	43.37%	32.82%	46.03%	39.10%	51.79%	48.95%	56.30%	43.49%
$X_P+X_{T_1}$	58.29%	63.60%	72.15%	60.17%	68.84%	61.49%	73.23%	66.74%
$X_P+X_{T_2}$	54.48%	49.11%	50.81%	69.43%	67.21%	70.20%	61.24%	64.10%
$X_P+X_{T_1}+X_{T_2}$	73.62%	83.13%	85.12%	81.96%	80.55%	79.97%	83.72%	85.23%

5.4.2 Necessary Training Data. In this experiment, we evaluate the performance of SugarMate with increasing numbers of training samples. Since the duration of measurements for each participant varies from 6 to 30 days, we use measurements of 5 to 25 days for training, and the rest for testing. Note that we keep the measurements for training but exclude them for testing if the duration of certain user's measurements is insufficient. For example, if the user's measurements last for 7 days, we use his measurement to evaluate the performance of using 5 days of training data, and test on the measurements of the remaining 2 days. However, when evaluating the performance with 10 days of training data, we only use his 7 days of measurements for training, but not for testing.

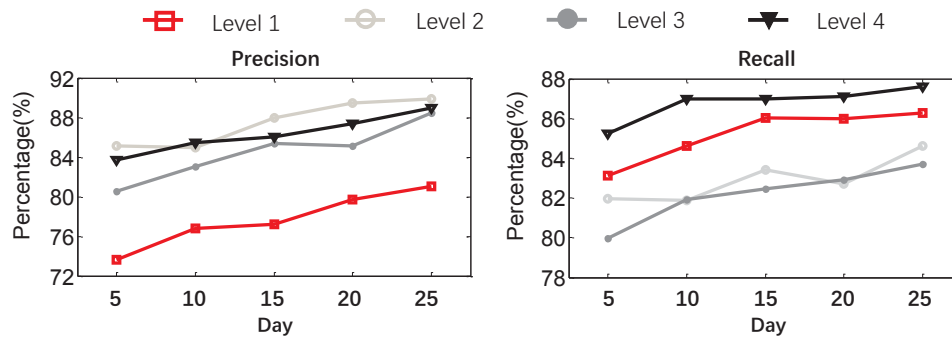


Fig. 11. Impact of increasing amount of training samples.

Fig. 11 illustrates the results for all 4 blood glucose levels. The results are averaged over all testing samples as in previous evaluations. As expected, the precisions and recalls of all 4 blood glucose levels improve smoothly with the increase of training samples. The results verify that the challenge (and our motivation to adopt a multi-division deep learning framework) is the lack of training data. Note that SugarMate is not a replacement to the current CGM devices, but rather a complement when CGM devices are uncomfortable or inconvenient to wear. Therefore, we envision the training dataset will grow gradually after wearing the CGM device multiple times (at least for diabetes patients), and the overall accuracy will also improve over time as a result.

5.4.3 Impact of Temporal Gaps. The blood glucose concentration is correlated with the previous blood glucose levels because of the control loop of the glucose metabolism [8, 23, 41]. Since SugarMate does not rely on the previous blood glucose value measured by CGM as an input, it is natural that the accuracy of SugarMate will degrade if there is a long gap between the training and the testing datasets.

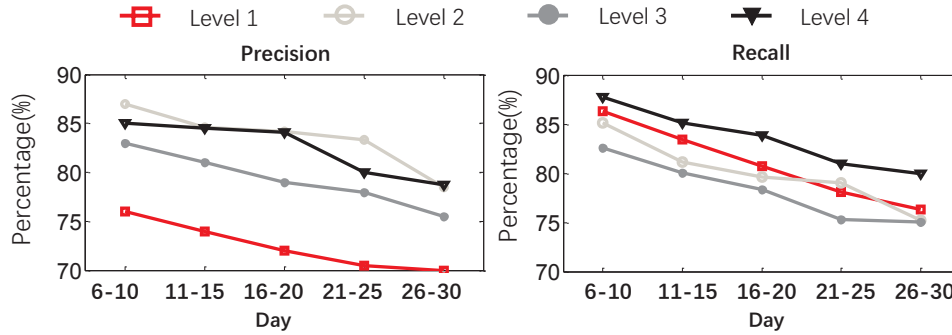


Fig. 12. Impact of temporal gaps between the training and testing datasets.

Fig. 12 plots the overall performance of training using the same 5 days of measurements, and testing on measurements collected on the 6-10th, 11-15th, 16-20th, 21-25th, and 26-30th days, respectively. As expected, both the precisions and recalls drop moderately with the increase of temporal gaps between the training and the testing datasets, with a maximum decrease of 6.73% and 7.02% in average precision and recall after 21-25 days. Note that SugarMate is not designed as a replacement to the commercial CGM devices, but rather a temporary alternative when CGM devices are uncomfortable or inconvenient to wear. From the results, we recommend users to put on the CGM devices to monitor the blood glucose at least once every three weeks. The data sampled by the CGM devices will automatically feed into SugarMate for model retraining. Although SugarMate still requires periodic re-calibration, it does not require users to continuously wear CGM devices.

5.4.4 Costs of Training Models. The resource cost of model learning in the learning phase is another important factor. We measure the CPU and memory costs of Md³RNN on our server (Inter(R) Core(TM) i7-4510U CPU @ 2.00 GHZ 2.60 GHZ; RAM: 8GB) and compare it with other baseline models in 5.3.2. Table 8 illustrates the results of 1000 iterative training.

Table 8. Resource costs of training models

Method	RF	GB	SVM	GP	HMM	mSVM	mGP	nDP-iHMM	hANN	Md ³ RNN
CPU(s)	19.5	9.5	20.1	26.4	29.5	165.8	216.2	319.1	208.6	259.9
RAM(MB)	84.2	84.0	188.4	213.9	112.1	1033.8	1428.5	1263.0	1181.4	1235.2

As is shown, the CPU and memory costs of share schemes are larger than those of the single learning algorithms. It meets our intuition that the share schemes are trained on the data from all users, while the single learning methods are only trained on the data of single person.

Since the personal blood glucose data is limited, the size of the learning data used by Md³RNN is not large. The training time of Md³RNN and the RAM cost are acceptable in practical usage, and its performance can be further optimized by the GPU version of Tensorflow.

Table 9. Profiles of smartphone

Type	CPU	RAM	ROM	Power Capacity	Operation System
Galaxy S6	8-cores 2.1 GHz	3 GB	32 GB	2550m Ah	Android 5.0
HTC Desire A6	8-cores 1.7 GHz	2 GB	16 GB	2600m Ah	Android 5.0
HUAWEI 4C	8-cores 1.2 GHz	2 GB	8 GB	3100m Ah	Android 4.4
LenovoK80M	4-cores 1.8 GHz	4 GB	64 GB	4000m Ah	Android 4.4

5.4.5 *Energy Overhead.* We evaluate the energy consumption of SugarMate on four popular types of smartphones used among our participants. Table 9 summarizes the smartphones used for evaluating the power consumption. Since SugarMate is a background service, we lock the screen and only leave SugarMate and a battery tracing application [45] running during the evaluation. The tracing application records the rest battery storage every two hours. We run SugarMate on fully charged smartphones and plot the remaining power over time in Fig. 13. In total, SugarMate and the Android OS consume about 10% energy of the battery every two hours, where roughly 40% of the power is consumed by SugarMate (see Fig. 14). Therefore, SugarMate takes about 4% of the total battery power every two hours given an inference rate of every 3 minutes. It is negligible and affordable for the daily usage.

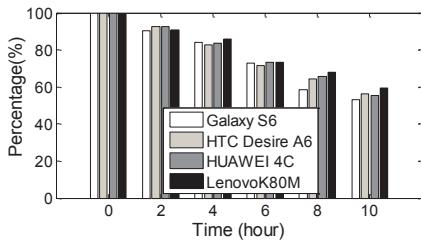


Fig. 13. Traces of remaining battery storage.

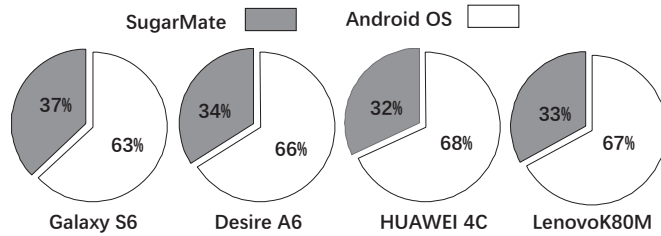


Fig. 14. Distribution of battery consumption.

6 PRELIMINARY USER STUDY

In addition to validating the effectiveness of Md³RNN, this section presents the feedbacks of users on the design and user experience of SugarMate.

6.1 Procedure

We distributed a semi-structured questionnaire to the 112 participants in Sec. 5. All of the participants answered the surveys, and each of them was paid about 20 USD (in the local currency) after the survey. We present the main results as follows⁴.

6.2 Results

6.2.1 *Operability of SugarMate.* In this part of the survey, the participants were asked to rate the overall operability of SugarMate as well as the three manual operations (food, drug and insulin intake) with three levels (*Inconvenient*, *Normal* and *Convenient*). The participants make comments on their practical operations. For example, the non-diabetic users who do not need to record drugs and insulin only rate the operation of food

⁴Note that most participants are not proficient in English. The original questionnaire was in the mouth tongue of the participants. The responses were carefully translated into English and presented in the results.

input, and the diabetic users who do not inject insulin are only report their comments on the food and drug inputs. They were also required to express their opinions on the overall operability.

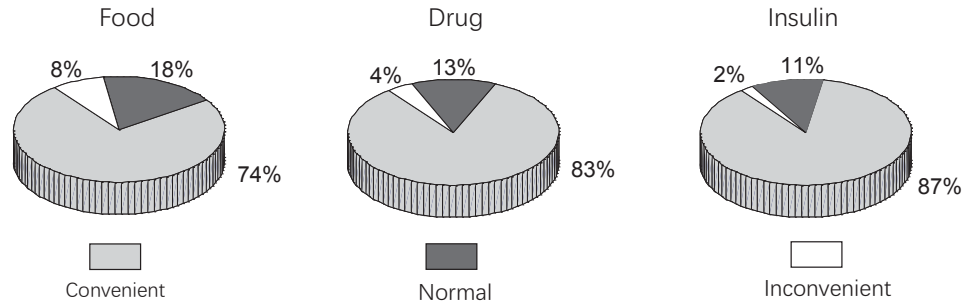


Fig. 15. Distributions of user rating on the operability of food, drug and insulin intake recording interfaces.

Fig. 15 illustrates the distributions of the user ratings on the operability of food, drug and insulin intake recording interfaces. The lowest rates are seen for manual recording of food intake (74% as *Convenient*). The ratings for manual recording of drug and insulin intake are slightly better. Overall, 78%, 18% and 4% of the participants rate SugarMate as *Convenient*, *Normal* and *Inconvenient*, respectively.

We select one representative comment for each level as follows:

- “This App is easy to learn and quick to use. The database of food, drugs and insulin is very comprehensive.” [Convenient, Type II diabetic user]
- “What I would like to see is a way to fill in the daily inputs by taking photos and speaking to. It would be more convenient to operate.” [Normal, Type I diabetic user]
- “I don’t have much time to record what I had for each meal using a scrolling menu. The App should be more personalized. For example, it should learn from my input history and provide search hints or automatic records.” [Inconvenient, non-diabetic user]

As expected, the manual recording of food, drug and insulin intake tends to be a burden for SugarMate as an intriguing application. Nevertheless, we have collected some constructive suggestions from the users. For instance, speech input instead of a scrolling menu might be a more welcome input mechanism. Intelligent search hints based on personal input history using recommendation algorithms [15] may help to improve the user experience when recording food intake, which is highly diversified and challenging to record automatically.

6.2.2 Benefits of SugarMate. In this part of the survey, each participant is asked to comment whether SugarMate is useful by rating SugarMate in three levels (*Instructive*, *Normal* and *Non-instructive*) and commenting in detail.

94% of the participants report SugarMate as *Instructive* and 6% report *Normal*. No participant rates SugarMate as *Non-instructive*. We also list two representative comments below.

- “SugarMate guides me to control my blood glucose and helps to live a healthier lifestyle. For instance, I always observe the blood glucose dynamics after meals. SugarMate helps to discover that my blood glucose rises a lot after I eat noodles, breads and dumplings, but stays relatively stable after eating meats and vegetables. Also I can learn about the impact of the drugs on my blood glucose. I’m happy to understand how my lifestyle affects my blood glucose and learn to make some adjustments in time. I have recommended SugarMate to six friends. They also care about their blood glucose levels.” [Instructive, Type II diabetic user]
- “I’m not diabetic but I feel I need to start tracking my blood glucose occasionally. This app has made it easy but still has room to improve. It presents me a short-term impact of food intake. However, I still care about

how to maintain a long-term healthy status. Specifically, it should provide some suggestions to help me prevent developing into diabetes.” [Normal, non-diabetic user]

In summary, users expect more functionalities of SugarMate such as recommending intensity and time of exercises and recommending food plans. In this regard, the personalized blood glucose dynamics records collected by SugarMate can be sent to experts to recommend personalized precautionary measures.

6.2.3 Willingness to use SugarMate. Finally, we are interested in whether the participants are willing to use SugarMate for blood glucose tracking despite the manual input of food, drug and insulin intake. In the questionnaire, we ask the following question: “Are you willing to use SugarMate instead of the CGM device at cost of manual recording of daily food, drug and insulin intake, as well as the periodic CGM re-calibration every three weeks ?” We also collect free-response comments as before.

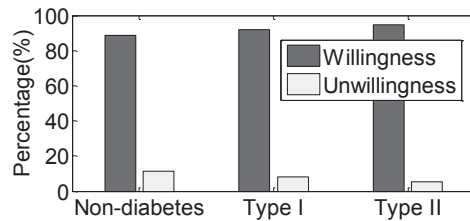


Fig. 16. Willingness to use SugarMate.

Fig. 16 illustrates the results. Over 90% diabetic users (both Type I and Type II) are willing to use SugarMate while non-diabetic users show a slightly lower willingness (85%). We also select two representative comments.

- “Manual records seem much more convenient compared to wearing the CGM. I feel very uncomfortable when wearing the CGM. It reminds me that I am a patient all the time. Wearing the CGM device for calibration once a month is acceptable. At least 3/4 of the time I am free from the CGM device.” [Willing, Type II diabetic user]
- “I am OK with the periodic CGM calibration, but the daily food recording is a pain for me. I always eat outside and easily forget what I have had after the meal. I’m not a patient. I don’t like to record everything I did simply to track my blood glucose.” [Unwilling, non-diabetic user]

From the feedbacks, it seems that most participants feel that continuous wearing of CGM devices is more uncomfortable. Even if periodic re-calibration is required, many diabetic users think it is acceptable. The manual recording of food intake is again the major concern, especially among non-diabetic participants. We will optimize the food recording function in our future work.

7 CONCLUSION

Inferring blood glucose levels is important to avoid health risks incurred by hyperglycemia and hypoglycemia. Commercial continuous blood glucose monitoring devices can be invasive and inconvenient to wear, which degrades the quality of life for diabetic patients and makes them inaccessible to non-diabetic people. We present SugarMate, a ubiquitous blood glucose level monitoring system using commodity smartphones. It measures important external factors that affect blood glucose concentration and adopts machine learning to infer blood glucose levels at a fine-grained time resolution. The core of SugarMate is an efficient blood glucose learning paradigm, Md³RNN, which (1) depicts complex glucose dynamics via a deep RNN model, (2) extracts generic feature representations with a grouped multi-division framework, and (3) preserves individual differences using personalized outputs. It tackles the sparsity and imbalance problem, which is the main hurdle of highly-accurate

personalized blood glucose level tracking. We deployed SugarMate to 112 users and collect measurements at least 6 days from each person. Evaluations show that Md³RNN outperforms the state-of-the-art methods on the blood glucose level inference. With fully automatic recording of external factors in the future, we envision SugarMate as a user-friendly and reliable complement for continuous blood glucose monitoring in daily life.

Acknowledgements: This work is supported by TBSI-Waveguide Joint Lab on Bio-Sensor and Medical Big Data. I also thank for the financial support by China Scholarship Council (CSC).

REFERENCES

- [1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, and others. 2016. TensorFlow: A system for large-scale machine learning. In *Proceedings of the USENIX Symposium on Operating Systems Design and Implementation*. USENIX Association, Berkeley, CA, USA, 265–283.
- [2] Edwin V Bonilla, Kian Ming Adam Chai, and Christopher KI Williams. 2008. Multi-task Gaussian process prediction. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., Red Hook, NY, USA, 153–160.
- [3] Razvan Bunescu, Nigel Struble, Cindy Marling, Jay Shubrook, and Frank Schwartz. 2013. Blood glucose level prediction using physiological models and support vector regression. In *Proceedings of IEEE International Conference on Machine Learning and Applications*, Vol. 1. IEEE Press, Piscataway, NJ, USA, 135–140.
- [4] Joslin Diabetes Center. 2017. Insulin A to Z: A Guide on Different Types of Insulin. (2017). <https://goo.gl/yIkOW2>.
- [5] Joslin Diabetes Center. 2017. Oral Diabetes Medications Summary Chart. (2017). <https://goo.gl/S0hSjX>.
- [6] Krzysztof Chalupka, Christopher KI Williams, and Iain Murray. 2013. A framework for evaluating approximation methods for Gaussian process regression. *Journal of Machine Learning Research* 14, Feb (2013), 333–350.
- [7] William L Clarke. 2005. The original Clarke error grid analysis (EGA). *Diabetes Technology & Therapeutics* 7, 5 (2005), 776–779.
- [8] Chiara Dalla Man, Robert A Rizza, and Claudio Cobelli. 2007. Meal simulation model of the glucose-insulin system. *Transactions on Biomedical Engineering* 54, 10 (2007), 1740–1749.
- [9] David L Duke. 2010. *Intelligent diabetes assistant: A telemedicine system for modeling and managing blood glucose*. PhD thesis. Carnegie Mellon University.
- [10] Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A Smith. 2015. Transition-based dependency parsing with stack long short-term memory. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing*. Association for Computational Linguistics, Stroudsburg PA, USA, 334–343.
- [11] Margaret Eckert-Norton and Susan Kirk. 2013. Non-diabetic hypoglycemia. *The Journal of Clinical Endocrinology & Metabolism* 98, 10 (2013), 39A–40A.
- [12] Theodoros Evgeniou and Massimiliano Pontil. 2004. Regularized multi-task learning. In *Proceedings of the tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM Press, New York, NY, USA, 109–117.
- [13] Philip Felig, Ali Cherif, Akira Minagawa, and John Wahren. 1982. Hypoglycemia during prolonged exercise in normal men. *New England Journal of Medicine* 306, 15 (1982), 895–900.
- [14] Jerome H Friedman. 2002. Stochastic gradient boosting. *Computational Statistics & Data Analysis* 38, 4 (2002), 367–378.
- [15] Xiaobin Fu, Jay Budzik, and Kristian J Hammond. 2000. Mining navigation history for recommendation. In *Proceedings of the ACM International Conference on Intelligent User Interfaces*. ACM Press, New York, NY, USA, 106–112.
- [16] Martha Funnell. 2016. Low Blood Glucose (Hypoglycemia). <https://goo.gl/SMeawu>. (2016).
- [17] Eleni Georga, Vasilios Protopappas, Alejandra Guillen, Giuseppe Fico, Diego Ardigo, Maria Teresa Arredondo, Themis P Exarchos, Demosthenes Polyzos, and Dimitrios I Fotiadis. 2009. Data mining for blood glucose prediction and knowledge discovery in diabetic patients: The METABO diabetes modeling and management system. In *Proceedings of IEEE Annual International Conference of the Engineering in Medicine and Biology Society*. IEEE Press, Piscataway, NJ, USA, 5633–5636.
- [18] Alex Graves, Marcus Liwicki, Horst Bunke, Jürgen Schmidhuber, and Santiago Fernández. 2008. Unconstrained on-line handwriting recognition with recurrent neural networks. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., Red Hook, NY, USA, 577–584.
- [19] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE Press, Piscataway, NJ, USA, 6645–6649.
- [20] DCCT Research Group and others. 1991. Epidemiology of severe hypoglycemia in the Diabetes Control and Complications Trial. *The American Journal of Medicine* 90, 4 (1991), 450–459.
- [21] Weixi Gu, Zheng Yang, Longfei Shanguan, Wei Sun, Kun Jin, and Yunhao Liu. 2014. Intelligent sleep stage mining service with smartphones. In *Proceedings of ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM Press, New York, NY, USA, 649–660.

- [22] Weixi Gu, Kai Zhang, Zimu Zhou, Ming Jin, Yuxun Zhou, Xi Liu, Costas J Spanos, Zuo-Jun Max Shen, Wei-Hua Lin, and Lin Zhang. 2017. Measuring fine-grained metro interchange time via smartphones. *Transportation Research Part C: Emerging Technologies* 81 (2017), 153–171.
- [23] Roman Hovorka, Valentina Canonico, Ludovic J Chassin, Ulrich Haueter, Massimo Massi-Benedetti, Marco Orsini Federici, Thomas R Pieber, Helga C Schaller, Lukas Schaupp, Thomas Vering, and others. 2004. Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes. *Physiological Measurement* 25, 4 (2004), 905.
- [24] Shingo Iwasaki, Junji Kozawa, Kenji Fukui, Hiromi Iwahashi, Akihisa Imagawa, and Ichihiro Shimomura. 2015. Coefficient of variation of R-R interval closely correlates with glycemic variability assessed by continuous glucose monitoring in insulin-depleted patients with type 1 diabetes. *Diabetes Research and Clinical Practice* 109, 2 (2015), 397–403.
- [25] Hui-xing Jia and Yujin Zhang. 2009. Fast Adaboost Training Algorithm by Dynamic Weight Trimming. *Chinese Journal of Computing* 32 (2009), 336–341.
- [26] Ming Jin, Han Zou, Kevin Weekly, Ruoxi Jia, Alexandre M Bayen, and Costas J Spanos. 2014. Environmental Sensing by Wearable Device for Indoor Activity and Location Estimation. In *Proceedings of IEEE Annual Conference of the Industrial Electronics Society*. IEEE Press, Piscataway, NJ, USA, 5369–5375.
- [27] Ronald Klein. 1995. Hyperglycemia and microvascular and macrovascular disease in diabetes. *Diabetes Care* 18, 2 (1995), 258–268.
- [28] Jan Koutnik, Klaus Greff, Faustino Gomez, and Juergen Schmidhuber. 2014. A clockwork rnn. In *Proceedings of International Conference on Machine Learning*. ACM Press, New York, NY, USA, 1863–1871.
- [29] Jennifer R. Kwapisz, Gary M. Weiss, and Samuel A. Moore. 2011. Activity Recognition Using Cell Phone Accelerometers. *SIGKDD Explorations Newsletter* 12, 2 (2011), 74–82.
- [30] Nicholas D Lane, Petko Georgiev, and Lorena Qendro. 2015. DeepEar: robust smartphone audio sensing in unconstrained acoustic environments using deep learning. In *Proceedings of ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM Press, New York, NY, USA, 283–294.
- [31] Nicholas D Lane, Emiliano Miluzzo, Hong Lu, Daniel Peebles, Tanzeem Choudhury, and Andrew T Campbell. 2010. A survey of mobile phone sensing. *IEEE Communications magazine* 48, 9 (2010), 140–150.
- [32] ED Lehmann and T Deutsch. 1992. A physiological model of glucose-insulin interaction in type 1 diabetes mellitus. *Journal of Biomedical Engineering* 14, 3 (1992), 235–242.
- [33] Andy Liaw and Matthew Wiener. 2002. Classification and regression by randomForest. *R news* 2, 3 (2002), 18–22.
- [34] Shenzhen Waveguide Optical Telecom Technology Inc. 2015. Continuous Glucose Monitor of WAVEGUIDER. (2015). <https://goo.gl/oAV0IE>.
- [35] Cindy Marling, Lijie Xia, Razvan Bunescu, and Frank Schwartz. 2016. Machine Learning Experiments with Noninvasive Sensors for Hypoglycemia Detection. In *Proceedings of IJCAI Workshop on Knowledge Discovery in Healthcare Data*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1 – 6.
- [36] Inc. MyFitnessPal. 2017. Calorie Counter - MyFitnessPal. (2017). <https://goo.gl/ETwrJS>.
- [37] Linh Lan Nguyen, Steven Su, and Hung T Nguyen. 2012. Identification of hypoglycemia and hyperglycemia in type 1 diabetic patients using ECG parameters. In *Proceedings of IEEE Annual International Conference of the Engineering in Medicine and Biology Society*. IEEE Press, Piscataway, NJ, USA, 2716–2719.
- [38] Kai Ni, Lawrence Carin, and David Dunson. 2007. Multi-task learning for sequential data via iHMMs and the nested Dirichlet process. In *Proceedings of the ACM International Conference on Machine Learning*. ACM Press, New York, NY, USA, 689–696.
- [39] Feiping Nie, Heng Huang, Xiao Cai, and Chris H Ding. 2010. Efficient and robust feature selection via joint L-2, 1-norms minimization. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., Red Hook, NY, USA, 1813–1821.
- [40] The Regents of the University of Michigan. 2012. Diabetes: Carbohydrate Food List. (2012). <https://goo.gl/dvV75f>.
- [41] Silvia Oviedo, Josep Vehí, Remei Calm, and Joaquim Armengol. 2016. A review of personalized blood glucose prediction strategies for T1DM patients. *International Journal for Numerical Methods in Biomedical Engineering* 33, 6 (2016), 6:1 – 6:21.
- [42] Barak A Pearlmutter. 1989. Learning state space trajectories in recurrent neural networks. *Neural Computation* 1, 2 (1989), 263–269.
- [43] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.
- [44] Kevin Plis, Razvan Bunescu, Cindy Marling, Jay Shubrook, and Frank Schwartz. 2014. A Machine Learning Approach to Predicting Blood Glucose Levels for Diabetes Management. In *Proceedings of AAAI Workshop on Modern Artificial Intelligence for Health Analytics*. AAAI Press, Palo Alto, CA, USA, 35 – 39.
- [45] PowerTutor.org. 2013. PowerTutor. <https://goo.gl/tiI38l>. (2013).
- [46] Lawrence Rabiner and B Juang. 1986. An introduction to hidden Markov models. *IEEE ASSP magazine* 3, 1 (1986), 4–16.
- [47] Carl Edward Rasmussen. 2004. Gaussian processes for machine learning. *Lecture Notes in Computer Science* 3176 (2004), 63–71.
- [48] Maximilian P Reymann, Eva Dorschky, Benjamin H Groh, Christine Martindale, Peter Blank, and Bjoern M Eskofier. 2016. Blood glucose level prediction based on support vector regression using mobile platforms. In *Proceedings of IEEE Annual International Conference of the*

- Engineering in Medicine and Biology Society*. IEEE Press, Piscataway, NJ, USA, 2990–2993.
- [49] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. 2012. Practical bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., Red Hook, NY, USA, 2951–2959.
 - [50] Sandra I Sobel, Peter J Chomentowski, Nisarg Vyas, David Andre, and Frederico GS Toledo. 2014. Accuracy of a novel noninvasive multisensor technology to estimate glucose in diabetic subjects during dynamic conditions. *Journal of Diabetes Science and Technology* 8, 1 (2014), 54–63.
 - [51] Santhisagar Vaddiraju, Diane J Burgess, Ioannis Tomazos, Faquir C Jain, and Fotios Papadimitrakopoulos. 2010. Technologies for continuous glucose monitoring: current problems and future promises. *Journal of Diabetes Science and Technology* 4, 6 (2010), 1540–1562.
 - [52] Lipo Wang. 2005. *Support vector machines: theory and applications*. Vol. 177. Springer Science & Business Media, New York, NY, USA.
 - [53] Sun-Chong Wang. 2003. Artificial neural network. In *Interdisciplinary computing in java programming*. Springer, New York, NY, USA, 81–100.
 - [54] Rachel A Whitmer, Andrew J Karter, Kristine Yaffe, Charles P Quesenberry, and Joseph V Selby. 2009. Hypoglycemic episodes and risk of dementia in older patients with type 2 diabetes mellitus. *The Journal of the American Medical Association* 301, 15 (2009), 1565–1572.
 - [55] Wikipedia. 2016. Blood glucose monitoring. (2016). <https://goo.gl/dW9Q6Z>.
 - [56] Wikipedia. 2017. Blood sugar. (2017). <https://goo.gl/ynpvO8>.
 - [57] Zheng Yang, Longfei Shangguan, Weixi Gu, Zimu Zhou, Chenshu Wu, and Yunhao Liu. 2014. Sherlock: Micro-environment sensing for smartphones. *IEEE Transactions on Parallel and Distributed Systems* 25, 12 (2014), 3295–3305.
 - [58] Yuxun Zhou, Ninghang Hu, Costas J Spanos, and others. 2016. Veto-Consensus Multiple Kernel Learning.. In *AAAI*. 2407–2414.
 - [59] Yuxun Zhou and Costas J Spanos. 2016. Causal meets submodular: Subset selection with directed information. In *Advances in Neural Information Processing Systems*. 2649–2657.

Received February 2017; revised May 2017