



Article

Meal and Physical Activity Detection from Free-Living Data for Discovering Disturbance Patterns of Glucose Levels in People with Diabetes

Mohammad Reza Askari ¹, Mudassir Rashid ¹, Xiaoyu Sun ², Mert Sevil ², Andrew Shahidehpour ¹, Keigo Kawaji ² and Ali Cinar ^{1,2,*}

¹ Department of Chemical and Biological Engineering, Illinois Institute of Technology, 10 W 33rd Street, Perlstein Hall, Suite 127, Chicago, IL 60616, USA; maskari@hawk.iit.edu (M.R.A.); mrashid3@iit.edu (M.R.); ashahide@hawk.iit.edu (A.S.)

² Department of Biomedical Engineering, Illinois Institute of Technology, Chicago, IL 60616, USA; xsun49@hawk.iit.edu (X.S.); msevil@hawk.iit.edu (M.S.); kkawaji@iit.edu (K.K.)

* Correspondence: cinar@iit.edu; Tel.: +1-312-567-3042

Abstract: Objective: The interpretation of time series data collected in free-living has gained importance in chronic disease management. Some data are collected objectively from sensors and some are estimated and entered by the individual. In type 1 diabetes (T1D), blood glucose concentration (BGC) data measured by continuous glucose monitoring (CGM) systems and insulin doses administered can be used to detect the occurrences of meals and physical activities and generate the personal daily living patterns for use in automated insulin delivery (AID). Methods: Two challenges in time-series data collected in daily living are addressed: data quality improvement and the detection of unannounced disturbances of BGC. CGM data have missing values for varying periods of time and outliers. People may neglect reporting their meal and physical activity information. In this work, novel methods for preprocessing real-world data collected from people with T1D and the detection of meal and exercise events are presented. Four recurrent neural network (RNN) models are investigated to detect the occurrences of meals and physical activities disjointly or concurrently. Results: RNNs with long short-term memory (LSTM) with 1D convolution layers and bidirectional LSTM with 1D convolution layers have average accuracy scores of 92.32% and 92.29%, and outperform other RNN models. The F1 scores for each individual range from 96.06% to 91.41% for these two RNNs. Conclusions: RNNs with LSTM and 1D convolution layers and bidirectional LSTM with 1D convolution layers provide accurate personalized information about the daily routines of individuals. Significance: Capturing daily behavior patterns enables more accurate future BGC predictions in AID systems and improves BGC regulation.

Keywords: recurrent neural networks; event detection; data preprocessing; outlier removal; type 1 diabetes



Citation: Askari, M.R.; Rashid, M.; Sun, X.; Sevil, M.; Shahidehpour, A.; Kawaji, K.; Cinar, A. Meal and Physical Activity Detection from Free-Living Data for Discovering Disturbance Patterns to Glucose Levels in People with Diabetes. *Biomedinformatics* **2022**, *2*, 297–317. <https://doi.org/10.3390/biomedinformatics2020019>

Academic Editor: Pentti Nieminen

Received: 23 April 2022

Accepted: 27 May 2022

Published: 1 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Time series data are widely used in many fields, and various data-driven modeling techniques are developed to represent the dynamic characteristics of systems and forecast the future behavior. The growing research in artificial intelligence has provided powerful machine learning (ML) techniques to contribute to data-driven model development. Real-world data provide several challenges to modeling and forecasting, such as missing values and outliers. Such imperfections in data can reduce the accuracy of ML and the models developed. This necessitates data preprocessing for the imputation of missing values, down- and up-sampling, and data reconciliation. Data preprocessing is a laborious and time-consuming effort since big data are usually stacked on a large scale [1]. When models are used for forecasting, the accuracy of forecasts improve if the effects of future possible

disturbances based on behavior patterns extracted from historical data are incorporated in the forecasts. This paper focuses on these two problems and investigates the benefits of preprocessing the real-world data and the performance of different recurrent neural network (RNN) models for detecting various events that affect blood glucose concentration (BGC) in people with type 1 diabetes (T1D). The behavior patterns detected are used for more accurate predictions of future BGC variations, which can be used for warnings and for increasing the effectiveness of automated insulin delivery (AID) systems.

Time series data captured in daily living of people with chronic conditions have many of these challenges to modeling, detection, and forecasting. Focusing on people with T1D, the medical objective is to forecast the BGC of a person with T1D and prevent the excursion of BGC outside a “desired range” (70–180 mg/dL) to reduce the probability of hypo- and hyperglycemia events. In recent years, the number of people with diabetes has grown rapidly around the world, reaching pandemic levels [2,3]. Advances in continuous glucose monitoring (CGM) systems, insulin pump and insulin pen technologies, and in novel insulin formulations has enabled many powerful treatment options [4–9]. The current treatment options available to people with T1D range from manual insulin injections to AID. Manual injection (insulin bolus) doses are computed based on the person’s characteristics and the properties of the meal consumed. Current AID systems necessitate the manual entry of meal information to give insulin boluses for mitigating the effects of meal on the BGC. A manual adjustment of the basal insulin dose and increasing the BGC target level and/or consumption of snacks are the options to mitigate the effects of physical activity. Some people may forget to make these manual entries and a system that can nudge them to provide appropriate information can reduce the extreme excursions in BGC. Commercially available AID systems are hybrid closed-loop systems, and they require these manual entries by the user. AID systems, also called artificial pancreas (AP), consist of a CGM, an insulin pump, and a closed-loop control algorithm that manipulates the insulin infusion rate delivered by the pump based on the recent CGM values reported [10–23]. More advanced AID systems that use a multivariable approach [10,24–26] use additional inputs from wearable devices (such as wristbands) to automatically detect the occurrence of physical activity and incorporate this information to the automated control algorithms for a fully automated AID system [27]. Most AID systems use model predictive control techniques that predict future BGC values in making their insulin dosing decisions. Knowing the habits of the individual AID user improves the control decisions since the prediction accuracy of the future BGC trajectories can explicitly incorporate the future potential disturbances to the BGC, such as meals and physical activities, that will occur with high likelihood during the future BGC prediction window [24,26]. Consequently, the detection of meal and physical activity events from historical free-living data of a person with T1D will provide useful information for decision making by both the individual and by the AID system.

CGM systems report subcutaneous glucose concentration to infer BGC with a sampling rate of 5 min. Self-reported meal and physical activity data are often based on diary entries. Physical activity data can also be captured by wearable devices. The variables reported by wearable devices may have artifacts, noise, missing values, and outliers. The data used in this work include only CGM values, insulin dosing information, and diary entries of meals and physical activities.

Analyzing long-term data of people with T1D indicates that individuals tend to repeat daily habitual behaviors. Figure 1 illustrates the probability of physical activity and meal (indicated as carbohydrate intake) events, either simultaneously or disjointly, for 15 months of self-reported CGM, meal, insulin pump, and physical activity data of individuals with T1D. Major factors affecting BGC variations usually occur at specific time windows and conditions, and some combinations of events are mutually exclusive. For example, insulin-bolusing and physical activity are less likely to occur simultaneously or during hypoglycemia episodes, since people do not exercise when their BGC is low. People may have different patterns of behavior during the work week versus weekends or holidays. Predicting the probabilities of exercise, meal consumption, and their concurrent

occurrence based on historical data using ML can provide important information on the behavior patterns for making medical therapy decisions in diabetes.

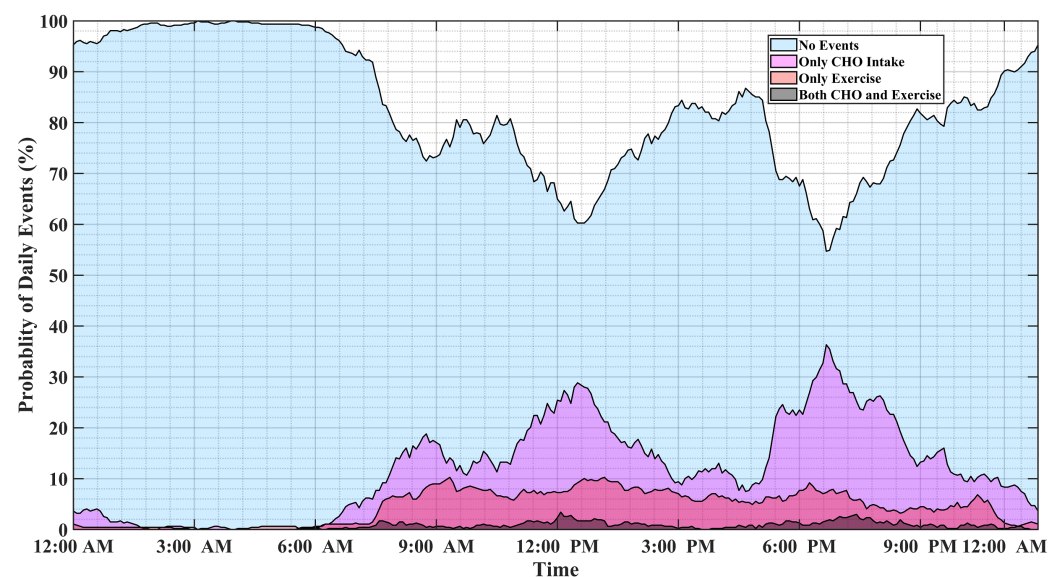


Figure 1. The probabilities of meal and physical activity events during one day obtained by analyzing 15 months of the pump–CGM sensor, meal, and physical activity data collected from a randomly selected person with T1D.

Motivated by the above considerations, this work develops a framework for predicting the probabilities of meal and physical activity events, including their independent and simultaneous occurrences. A framework is built to handle the inconsistencies and complexities of real-world data, including missing data, outlier removal, feature extraction, and data augmentation. Four different recurrent neural network (RNN) models are developed and evaluated for estimating the probability of events causing large variations in BGC. The advent of deep neural networks (NNs) and their advances have paved the way for processing and analyzing various types of information, namely: time-series, spatial, and time-series-spatial data. Long short-term memory (LSTM) NN models are specific sub-categories of recurrent NNs introduced to reduce the computational burden of storing information over extended time intervals [28,29]. LSTMs take advantage of nonlinear dynamic modeling without knowing time-dependency information in the data. Moreover, their multi-step-ahead prediction capability makes them an appropriate choice for detecting upcoming events and disturbances that can deteriorate the accuracy of model predictions.

The main contributions of this work are the development of NN models capable of estimating the occurrences of meals and physical activities without requiring additional bio-signals from wearable devices, and the integration of convolution layers with LSTM that enable the NN to accurately estimate the output from glucose–insulin input data. The proposed RNN models can be integrated with the control algorithm of an AID system to enhance its performance by readjusting the conservativeness and aggressiveness of the AID system.

The remainder of this paper is organized as follows: the next section provides a short description of the data collected from people with T1D. The preprocessing step, including outlier removal, data imputation, and feature extraction is presented in Section 3. Section 4 presents various RNN configurations used in this study. A case study with real-world data and a discussion of the results are presented in Section 5 and Section 6, respectively. Finally, Section 6 provides the conclusions.

2. Free-Living, Self-Reported Dataset of People with T1D

A total of 300 self-collected T1D datasets were made available for research, and each dataset represents a unique individual. Among all of the datasets, 50 T1D datasets include CGM-sensor–insulin-pump recordings and exercise information such as the time, type, and duration of physical activity recorded from either open or closed-loop insulin-pump–sensor data. Meal information is reported as the amount of carbohydrates (CHO) consumed in the meal as estimated by the subject. An over or underestimation of CHO in meals is common.

The subjects with T1D selected for this study used insulin-pump–CGM-sensor therapy for up to two years, and some of them have lived with diabetes for more than fifty years. Tables 1 and 2 summarize the demographic information of the selected subjects and the definition of the variables collected, respectively. Separate RNN models were developed for each person in order to capture personalized patterns of meal consumption and physical activity.

Table 1. The general demographic information of 11 subjects with T1D and the durations of recorded samples.

Subject	Gender	Age	Duration of Data ¹	Missing Samples (%)	Max Gap Size ²
1	M	36	283 days	12.36%	273
2	M	33	368 days	7.11%	71
3	F	72	280 days	1.10%	28
4	M	43	468 days	10.03%	435
5	F	52	655 days	4.91%	233
6	F	26	206 days	14.45%	107
7	M	51	278 days	6.12%	34
8	-	41	390 days	8.87%	177
9	-	42	279 days	19.70%	311
10	M	27	695 days	14.32%	571
11	F	35	413 days	8.97%	147

¹ The duration of data is calculated after imputation of missing data and counting gaps between samples.

² Number of samples, sampling time 5 min.

Table 2. The name and the definition of measured variables.

Variable/Symbol	Definition	Units
CGM	Continuous glucose monitoring values sampled every five minutes	mmol/L
Smbg	Self-monitored BGC for sensor calibration	mmol/L
Rate (Ins_{Basal})	The basal insulin rate	unit/h
Bolus (Ins_{Bolus})	The actual delivered amount of normal bolus insulin	unit
Time	UTC time stamp	Format: yyyy-mm-yy hh:mm:ss
Duration	The actual duration of a suspend, basal, or dual/square bolus	milliseconds
Activity.name	The type of physical activity	-
Activity.duration (AD)	The duration of a physical activity	milliseconds
Distance.value (DV)	The value of the distance traveled	miles
Energy.value (EV)	The amount of energy spent during activity	kilocalories
Nutrition.carbohydrate (CHO)	The carbohydrates entered in a health kit food entry	grams

3. Data Preprocessing

This is a computational study for the development of detection and classification of infrequent events (eating, exercising) that affect the main variable of interest in people with diabetes: their blood glucose concentrations. It is based on data collected from patients in free living; hence, it contains many windows of data with missing values and outliers. Using real-world data for developing models usually has numerous challenges: (i) the datasets can be noisy and incomplete; (ii) there may be duplicate CGM samples in

some of the datasets; (iii) inconsistencies exist in the sampling rate of CGM and insulin values; (iv) gaps in the time and date can be found due to insulin pump or CGM sensor disconnection. Therefore, the datasets need to be preprocessed before using them for model development.

3.1. Sample Imputation

Estimating missing data is an important step before analyzing the data [30]. Missing data are substituted with reasonable estimates (imputation) [31]. In dealing with time-series data such as CGM, observations are sorted according to their chronological order. Therefore, the variable “Time”, described in Table 2, is converted to “Unix time-stamp”, samples are sorted in ascending order of “Unix time-stamp”, and gaps without observations are filled with pump–sensor samples labeled as “missing values”.

Administered basal insulin is a piecewise constant variable and its amount is calculated by the AID system or by predefined insulin injection scenarios. Applying a simple forward or backward imputation for basal insulin with gaps in duration lasting a maximum of two hours gives reasonable reconstructed values for the missing observations. Gaps lasting more than two hours in missing recordings are imputed with basal insulin values recorded in the previous day at the same time, knowing that insulin injection scenarios usually follow a daily pattern [32].

The variable “Bolus” is a sparse variable (usually nonzero only at times of meals) and its missing samples were imputed with the median imputation approach, considering that the bolus injection policy is infrequently altered. Similarly, missing recordings of variables “Nutrition.carbohydrate”, “Smbg”, “Duration”, “Activity.duration”, and “Distance.value” were imputed with the median strategy. A multivariate strategy that uses CGM, total injected insulin, “Nutrition.carbohydrate”, the “Energy.value”, and “Activity.duration” was employed to impute missing CGM values.

This choice of variables has to do with the dynamic relationship between CGM and the amount of carbohydrate intake, the duration and the intensity of physical activity, and the total injected insulin. Estimates of missing CGM samples were obtained by performing probabilistic principal component analysis (PPCA) on the lagged matrices of the CGM data. PPCA is an extension of principal component analysis, where the Gaussian conditional distribution of the latent variables is assumed [33]. This formulation of the PPCA facilitates tackling the problem of missing values in the data through the maximum likelihood estimation of the mean and variance of the original data. Before performing PPCA on the feature variables, the lagged array of each feature variable, $\mathcal{X}_{k,j}, k \in \{CGM, Ins, CHO, EV, AD\}$, at the j th sampling index was constructed from the past two hours of observations as:

$$\begin{aligned} \mathcal{X}_{k,j} &= [X_{k,j}, X_{k,j-1} \dots X_{k,j-24}]_{1 \times 25}, \quad k \in \{CGM, Ins, CHO, EV, AD\} \\ X_j &= [\mathcal{X}_{1,j}, \dots, \mathcal{X}_{M,j}]^T, \quad X = [X_1, \dots, X_N]_{M \times N} \end{aligned} \quad (1)$$

For an observed set of feature variables X_j , let $\mathcal{T}_j = [\mathcal{T}_{1,j}, \dots, \mathcal{T}_{q,j}]^T$ be its q -dimensional ($q \leq M$) Gaussian latent transform [34] such that

$$X_{i,j} = W_i \mathcal{T}_j + \mu_i + \epsilon_{i,j} \quad (2)$$

where $W_i = [W_{i,1}, \dots, W_{i,q}] \in \mathbb{R}^q$ and $\underline{\mu} = [\mu_1, \dots, \mu_M]^T \in \mathbb{R}^M$ represent the i th row of the loading matrix $W \in \mathbb{R}^{M \times q}$ and mean value of the data. $\epsilon_{i,j} \in \mathbb{R}$ is also the measurement noise with the probability distribution

$$p(\epsilon_{i,j} | \sigma^2) = \mathcal{N}(\epsilon_{i,j} | 0, \sigma^2). \quad (3)$$

Based on the Gaussian distribution assumption of $\mathcal{T}_j$ and the Gaussian probability distribution of $\epsilon_{i,j}$, one can deduce that

$$\begin{cases} p(\mathcal{T}_j) = \mathcal{N}(\mathcal{T}_j|0, I_q) \\ p(X_{i,j}|\mu_i, W_i, \sigma^2) = \mathcal{N}(X_{i,j}|\mu_i, W_i W_i^T + \sigma^2) \\ p(X_{i,j}|\mathcal{T}_j, \mu_i, W_i, \sigma^2) = \mathcal{N}(X_{i,j}|\mu_i + W_i \mathcal{T}_j, W_i W_i^T + \sigma^2) \end{cases} \quad (4)$$

The joint probability distribution $p(X_{i,j}, \mathcal{T}_j, \mu_i, W_i, \sigma^2)$ can be derived from (4) and Bayes' joint probability rule as

$$p(X_{i,j}, \mathcal{T}_j, \mu_i, W_i, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{\frac{M}{2}}} \exp\left(\frac{(X_{i,j} - W_i \mathcal{T}_j - \mu_i)^2}{-2\sigma^2}\right) \frac{1}{(2\pi)^{\frac{q}{2}}} \exp\left(\frac{-\mathcal{T}_j^T \mathcal{T}_j}{2}\right) \quad (5)$$

Define the set $\eta = \{(i, j) | 1 \leq i \leq M, 1 \leq j \leq N, X_{i,j} \neq \text{NaN}\}$. The log-likelihood of the joint multivariate Gaussian probability distribution of (5) is calculated over all available observations as

$$\begin{aligned} \ln(p(X_{i,j}, \mathcal{T}_j|\mu_i, W_i, \sigma^2)) &= \sum_{i,j \in \eta} [\ln(p(X_{i,j}|\mathcal{T}_j, \mu_i, W_i, \sigma^2)) + \ln p(\mathcal{T}_j)] \\ &= \sum_{i,j \in \eta} \sum \left[-\frac{M}{2} \ln(2\pi\sigma^2) - \frac{q}{2} \ln(2\pi) - \frac{(X_{i,j} - W_i \mathcal{T}_j - \mu_i)^2}{2\sigma^2} - \frac{\mathcal{T}_j^T \mathcal{T}_j}{2} \right] \end{aligned} \quad (6)$$

where the log-likelihood (6) is defined for all available observations $X_{i,j}, i, j \in \eta$. By applying the expectation operation with respect to the posterior probability distribution over all latent variables $\mathcal{T}_j, j \in \eta_i$, where $\eta_i = \{j | 1 \leq j \leq N, X_{i,j} \neq \text{NaN}\}$, (6) becomes

$$\begin{aligned} \mathbb{E}\{\mathcal{L}\} &= - \sum_{i,j \in \eta} \sum \left[\frac{M}{2} \ln(\sigma^2) + \frac{1}{2} \mathbb{E}\{\mathcal{T}_j^T \mathcal{T}_j\} + \frac{1}{2\sigma^2} (X_{i,j} - \mu_i)^2 \right. \\ &\quad \left. - \frac{1}{\sigma^2} \mathbb{E}\{\mathcal{T}_j^T\} W_i^T (X_{i,j} - \mu_i) + \frac{1}{2\sigma^2} \mathbb{E}\{\mathcal{T}_j^T \mathcal{T}_j\} W_i W_i^T \right] \end{aligned} \quad (7)$$

Maximizing (7) is feasible by setting all partial derivatives $\frac{\partial \mathbb{E}\{\mathcal{L}\}}{\partial \sigma^2}$, $\frac{\partial \mathbb{E}\{\mathcal{L}\}}{\partial \mu_i^2}$, and $\frac{\partial \mathbb{E}\{\mathcal{L}\}}{\partial W_i^2}$, $i = 1, \dots, M, j = 1, \dots, N$ to zero [34].

$$\begin{aligned} \text{Cvar}_{\mathcal{T}_j} &= \frac{\sigma^2}{(\sigma^2 I_q + \sum_{i \in \eta_j} W_i W_i^T)} \\ \mu_{\mathcal{T}_j} &= \frac{\text{Cvar}_{\mathcal{T}_j}}{\sigma^2} \sum_{i \in \eta_j} W_i^T (X_{i,j} - \mu_i) \\ \mu_i &= \frac{1}{|\eta_i|} \sum_{j \in \eta_i} [X_{i,j} - W_i \mu_{\mathcal{T}_j}] \\ W_i &= \frac{1}{\sum_{j \in \eta_i} [\mu_{\mathcal{T}_j} \mu_{\mathcal{T}_j}^T + \text{Cvar}_{\mathcal{T}_j}]} \sum_{j \in \eta_i} \mu_{\mathcal{T}_j} (X_{i,j} - \mu_i) \\ \sigma^2 &= \frac{1}{|\eta|} \sum_{i,j \in \eta} [(X_{i,j} - W_i \mu_{\mathcal{T}_j} - \mu_i)^2 + W_i \text{Cvar}_{\mathcal{T}_j} W_i^T] \end{aligned} \quad (8)$$

Parameters μ_i, σ^2 , and W_i in (8) are updated recursively until they converge to their final values. The final estimation of missing CGM samples is obtained by performing a diagonal averaging of the reconstructed lagged matrix $\hat{X} \in \mathbb{R}^{M \times N}$ over rows/columns filled with CGM values. Long gaps in CGM recordings might exist in the data, and imputing their values causes problems in accuracy and reliability. Therefore, CGM gaps

of no more than twenty-five consecutive missing samples (approximately two hours) are imputed by PPCA.

3.2. Outlier Removal

Signal reconciliation and outlier removal are necessary to avoid misleading interpretation of data and biased results, and to improve the quality of CGM observations. As a simple outlier removal approach for a variable with Gaussian distribution, observations outside ± 2.72 standard deviations from the mean, known as inner Tukey fences, can be labeled as outliers and extreme values [35]. The probability distribution of the CGM data shows a skewed distribution compared to the Gaussian probability distribution. Thus, labeling samples as outliers only based on their probability of occurrence is not the proper way of removing extreme values from the CGM data since it can cause a loss of useful CGM information, specifically during hypoglycemia ($CGM < 70$ mg/dL) and hyperglycemia ($CGM > 180$ mg/dL) events. As another alternative, extreme values and spikes in the CGM data can be labeled from the prior knowledge and by utilizing other feature variables, namely: “Smbg”, “Nutrition.carbohydrate”, “Bolus”, and “Activity.duration”. Algorithm 1 is proposed to remove outliers from CGM values. Usually, BGC is slightly different from the recordings of the CGM signal because of the delay between BGC and the subcutaneous glucose concentration measured by the CGM device and sensor noise. The noisy signal can deteriorate the performance of data-driven models. Therefore, Algorithm 2, which is based on eigendecomposition of the Hankel matrix of CGM values, is used to reduce the noise in the CGM recordings.

Algorithm 1 Outlier rejection from CGM readings

```

1: procedure OUTLIERREJECTION( $CGM, Smbg, CHO, AD, Ins_{Bolus}$ )
2:   for  $i = 1 : N$  do ▷ Removing samples outside of the calibration range
3:     if  $CGM_k > 400$  mg/dL or  $CGM_k < 0$  mg/dL then
4:        $CGM_k \leftarrow NaN$ 
5:     end if
6:   end for
7:   for  $i = 2 : N$  do
8:      $\Delta CGM_k \leftarrow CGM_k - CGM_{k-1}$ 
9:     if  $\Delta CGM_k > 30$  mg/dL & all ( $\{CHO_k, \dots, CHO_{k-9}\} == 0$ ) then
10:       $CGM_k \leftarrow NaN$ 
11:    end if
12:    if  $\Delta CGM_k < 30$  mg/dL & all ( $\{Ins_{Bolus,k}, \dots, Ins_{Bolus,k-6}\} == 0$ ) then
13:       $CGM_k \leftarrow NaN$ 
14:    end if
15:    if  $\Delta CGM_k < 30$  mg/dL & all ( $\{AD_k, \dots, AD_{k-6}\} == 0$ ) then
16:       $CGM_k \leftarrow NaN$ 
17:    end if
18:    if  $Smbg_k \neq NaN$  &  $CGM_k \neq NaN$  &  $abs(Smbg_k - CGM_k) > 18$  mg/dL then
19:       $CGM_k \leftarrow NaN$ 
20:    end if
21:  end for
22:  return  $CGM$ 
23: end procedure

```

Algorithm 2 Smoothing CGM recordings

```

1: procedure CGMDENOISING(CGM)                                ▷ Smoothing CGM recordings
2:    $Q_i = [CGM_d, \dots, CGM_{d+q_i-1}]$           ▷  $Q_i \in \mathbb{R}^{q_i}$  is  $i$ th consecutive CGM recordings
3:    $q_i \leftarrow |Q_i|, p_i \leftarrow \text{floor}(\frac{q_i}{2}), w_i \leftarrow q_i - p_i + 1$ 
4:    $[U_i, S_i, V_i] = \text{SVD}(A_i)$                 ▷  $A_i \in \mathbb{R}^{w_i \times p_i}$  is the Hankel matrix made of  $Q_i$ 
5:    $\hat{S}_i \leftarrow \text{zeros}(p_i, p_i)$ 
6:    $\eta \leftarrow \frac{\text{cumsum}([s_1, \dots, s_{p_i}])}{\text{sum}([s_1, \dots, s_{p_i}])}$           ▷  $s_j > 0$  are eigenvalues of  $S_i$  in descending order
7:   for  $j=1:p_i$  do
8:     if  $\eta_j > 0.95$  then
9:        $\hat{S}_i(j, j) \leftarrow 0$ 
10:    else
11:       $\hat{S}_i(j, j) \leftarrow S_i(j, j)$ 
12:    end if
13:  end for
14:   $\hat{A}_i = U_i \hat{S}_i V_i^T$ 
15:   $\hat{Q}_i \leftarrow \text{Diagonalaveraging}(\hat{A}_i)$           ▷  $\hat{Q}_i = [C\hat{G}M_d, \dots, C\hat{G}M_{d+q_i-1}]$ 
16:  return  $C\hat{G}M$ 
17: end procedure

```

3.3. Feature Extraction

Converting raw data into informative feature variables or extracting new features is an essential step of data preprocessing. In this study, four groups of feature variables, including frequency domain, statistical domain, nonlinear domain, and model-based features, were calculated and added to each dataset to enhance the prediction power of models. The summarized description of each group of features and the number of past samples required for their calculation are listed in Table 3.

A qualitative trend analysis of variables can extract different patterns caused by external factors within a specified time [36,37]. A pairwise multiplication of the sign and magnitude of the first and second derivatives of CGM values indicates the carbohydrate intake [38,39], exogenous insulin injection, and physical activity. Therefore, the first and second derivatives of CGM values, calculated by the fourth-order backward difference method, were added as feature variables. The sign and magnitude product of the first and second derivatives of CGM, their covariance, Pearson correlation coefficient, and Gaussian kernel similarity were extracted. Statistical feature variables, e.g., mean, standard deviation, variance, skewness, etc., were obtained from the specified time window of CGM values. Similar to the first and second derivatives of CGM values, a set of feature variables, including covariance and correlation coefficients, from pairs of CGM values and derivatives was extracted and augmented to the data.

As a result of the daily repetition in the trends of CGM and glycemic events and the longer time window of CGM values, samples collected during the last twenty-four hours were used for frequency-domain feature extraction. Therefore, magnitudes and frequencies of the top three dominant peaks in the power spectrum of CGM values, conveying past long-term variation of the BGC, were included in the set of feature maps.

Table 3. The type and definition of the extracted feature variables and the length of time window required for their calculations.

Domain	Feature Description	No. of Required Samples
Time	First derivative calculated by 4th backward differences	5
	Second derivative calculated by 4th backward differences	6
Nonlinear	Sign-product of the 1st and the 2nd derivatives	6
	Magnitude-product of the 1st and the 2nd derivatives	6
Statistical	Statistical measures, namely mean, variance, median, etc., of windowed CGM values	24
	Pair-wise covariance and correlation coefficient between CGM and its 1st and 2nd derivatives	24
Frequency	The magnitudes and frequencies of three dominant peaks in the power spectrum of CGM	288
Model-based	Plasma insulin concentration (PIC) and gut absorption rate (U_g) [40,41].	1

The plasma insulin concentration (PIC) is another feature variable that informs about the carbohydrate intake information and exogenous insulin administration. PIC accounts for the accumulation of subcutaneously injected insulin within the bloodstream, which is gradually consumed by the body to enable the absorption of carbohydrates released from the gastrointestinal track to various cells and tissues. Usually, dynamic physiological models are used to describe and model the glucose and insulin concentration dynamics in diabetes. The main idea of estimating PIC from physiological models stems from predicting the intermediate state variables of physiological models by designing a state observer and utilizing the total infused insulin and carbohydrate intake as model inputs, and CGM values as the output of the model [40–42]. In this work, the estimation of the PIC and glucose appearance rate were obtained from a physiological model known as Hovorka’s model [43]. Equation (9) presents this nonlinear physiological (compartment) model:

$$\begin{aligned}
 \frac{dS_1(t)}{dt} &= Ins(t) - \frac{S_1(t)}{t_{max,I}} \\
 \frac{dS_2(t)}{dt} &= \frac{S_1(t)}{t_{max,I}} - \frac{S_2(t)}{t_{max,I}} \\
 \frac{dI(t)}{dt} &= \frac{S_2(t)}{t_{max,I}V_I} - K_e I(t) \\
 \frac{dx_1(t)}{dt} &= k_{b,1}I(t) - k_{a,1}x_1(t) \\
 \frac{dx_2(t)}{dt} &= k_{b,2}I(t) - k_{a,2}x_2(t) \\
 \frac{dx_3(t)}{dt} &= k_{b,3}I(t) - k_{a,3}x_3(t) \\
 \frac{dQ_1(t)}{dt} &= U_g(t) - F_{0,1}^c(t) - F_R(t) - x_1(t)Q_1(t) + k_{12}Q_2(t) + EGP_0(1 - x_3(t)) \\
 \frac{dQ_2(t)}{dt} &= x_1(t)Q_1(t) - (k_{12} + x_2(t))Q_2(t) \\
 \frac{dG_{sub}(t)}{dt} &= \frac{1}{\tau} \left(\frac{Q_1(t)}{V_g} - G_{sub}(t) \right)
 \end{aligned} \tag{9}$$

Model (9) comprises four sub-models, describing the action of insulin on glucose dynamics, the insulin absorption dynamics, plasma–interstitial-tissue glucose concentration dynamics, and the blood glucose dynamics. The state variables of (9), the nominal values of the parameters, and their units are listed in Table 4 [43].

Table 4. The description of variables and parameters and the nominal values of parameters in Hovorka’s model [43].

Variable/Parameter	Description	Value/Unit
$S_1(t), S_2(t)$	Two-compartment chain representing absorption of subcutaneously administered short-acting insulin	mU
$Ins(t)$	Subcutaneously infused insulin	mU min ^{−1}
$I(t)$	Plasma insulin concentration (PIC)	mU L ^{−1}
$x_1(t)$	The remote effect of insulin on glucose distribution	min ^{−1}
$x_2(t)$	The remote effect of insulin on glucose disposal	min ^{−1}
$x_3(t)$	The remote effect of insulin on endogenous glucose production (EGP)	min ^{−1}
$Q_1(t)$	The mass of glucose in accessible compartments	mmol
$Q_2(t)$	The mass of glucose in non-accessible compartments	mmol
$G_{sub}(t)$	Measurable subcutaneous glucose concentration	mmol L ^{−1}
$U_G(t)$	Gut absorption rate	mmol min ^{−1}
K_e	The fractional elimination rate of PIC	0.138 min ^{−1}
$k_{a,1}$	The deactivation rate constants	0.006 min ^{−1}
$k_{a,2}$		0.06 min ^{−1}
$k_{a,3}$		0.03 min ^{−1}
S_{ID}^f	The sensitivity of insulin disposal	0.00082 L min ^{−1} mU ^{−1}
S_{IT}^f	The sensitivity of insulin distribution	0.00512 L min ^{−1} mU ^{−1}
S_{IE}^f	The sensitivity of EGP	0.052 L mU ^{−1}
$k_{b,1}$	The activation rate constants	$k_{a,1} \times S_{IT}^f$
$k_{b,2}$		$k_{a,2} \times S_{ID}^f$
$k_{b,3}$		$k_{a,3} \times S_{IE}^f$
EGP_0	EGP extrapolated to zero insulin concentration	0.0161 mmol kg ^{−1} min ^{−1}
k_{12}	The transfer rate constant from the non-accessible to the accessible compartment	0.066 min ^{−1}
τ	The time constant of subcutaneous glucose concentration dynamic	min
V_g	The glucose distribution volume in the accessible compartment	0.16 × BW(L)
V_I	The insulin distribution volume in the accessible compartment	0.12 × BW(L)
$F_R(t)$	The renal glucose clearance above the glucose threshold of 9 mmol L ^{−1}	$\begin{cases} 0.003(G_{sub} - 9), G_{sub} \geq 9 \\ 0, G_{sub} < 9 \end{cases}$
F_{01}	Non-insulin-dependent glucose flux	0.0097 mmol kg ^{−1} min ^{−1}
$F_{0,1}^c(t)$	The total non-insulin-dependent glucose flux (mmol min ^{−1})	$\begin{cases} F_{01}, G_{sub} \geq 4.5 \\ F_{01}(G_{sub}/4.5), G_{sub} < 4.5 \end{cases}$

Body weight has a significant effect on the variations in the PIC and other state variables as it is used for determining the amount of exogenous insulin to be infused. Although estimating body weight as an augmented state variable of the insulin-CGM model is an effective strategy to cope with the problem of unavailable demographic information, estimating body weight from the total amount of daily administered insulin is a more reliable approach. As reported in various studies, the total daily injected insulin can have a range of 0.4–1.0 units kg^{−1} day^{−1} [44–46]. A fair estimation of body weight can be obtained by calculating the most common amount of injected basal/bolus insulin for each subject and using a conversion factor of 0.5 units kg^{−1} day^{−1} as a rule of thumb to estimate the body weight.

The insulin–glucose dynamics (9) in discrete-time format are given by

$$\begin{aligned} X'_{k+1} &= f'(X'_k, U_k) + G_k \omega_k, & \omega_k &\approx N(0, Q) \\ Y'_k &= h'(X'_k) + v_k, & v_k &\approx N(0, R) \end{aligned} \quad (10)$$

where $X'_k = [S_{1,k}, S_{2,k}, I_k, x_{1,k}, x_{2,k}, x_{3,k}, Q_{1,k}, Q_{2,k}, G_{sub,k}, t_{max,I,k}, k_{e,k}, U_{G,k}] \in \mathcal{R}^{n_x}$ denotes the extended state variables and U_k is the total injected exogenous insulin. Symbols ω_k and v_k denote zero-mean Gaussian random process and measurement noises (respectively),

representing any other uncertainty and model mismatch that are not taken into account. Further, $Q \in \mathcal{R}^{n_x \times n_x}$ and $R \in \mathcal{R}$ represent the positive definite system uncertainty and measurement noise covariance matrices, respectively.

Tracking the dynamics of the internal state variables of the model (10) is feasible by using a class of sequential Monte Carlo algorithms known as particle filters. A generic form of the particle filter algorithm proposed by [47] with an efficient adaptive Metropolis–Hastings resampling strategy developed in [48] was employed to predict the trajectory of the PIC and other state variables. In order to avoid any misleading state estimations, each state variable was subjected to a constraint to maintain all estimations within meaningful intervals [41].

3.4. Feature Selection and Dimensionality Reduction

Reducing the number of redundant feature variables lowers the computational burden of their extraction and hinders over-parameterized modeling. In this work, a two-step feature selection procedure was used to obtain the optimal subset of feature variables that boost the efficiency of the classifier the most. In the first step, the deviance statistic test was performed to filter out features with low significance (p -value > 0.05). In the second step, the training split of all datasets was used in the wrapper feature selection strategy to maximize the accuracy of the classifier in estimating the glycemic events. A sequential floating forward selection (SFFS) approach [49] was applied on a random forest estimator with thirty decision tree classifiers with a maximum depth of six layers to sort out features with the most predictive power in descending order. Consequently, the top twenty feature variables with the highest contribution to the classification accuracy enhancement were used for model development.

4. Detection and Classification Methods

Detecting the occurrence of events causing large glycemic variations requires solving a supervised classification problem. Hence, all samples required labeling using the information provided in the datasets, specifically using variables “Activity.duration” and “Nutrition.carbohydrate”. In order to determine the index sets of each class, let N be the total number of samples and $T(k) = \text{ceil}(AD(k)/(3 \times 10^5))$ be the sample duration of physical activity at each sampling time k . Define sets of sample indexes as:

$$\begin{cases} \text{Label.Index}_{\{1,1\}} := \{i | k \leq i \leq k + T(k) - 1, \quad k = 1, \dots, N, \quad T(k) \neq 0, \\ \quad \text{Nutrition.carbohydrate}(j) \neq 0, \quad j = k + 1, \dots, k + T(k)\} \\ \text{Label.Index}_{\{0,1\}} := \{i | k \leq i \leq k + T(k) - 1, \quad k = 1, \dots, N, \quad T(k) \neq 0, \\ \quad \text{Nutrition.carbohydrate}(j) = 0, \quad j = k + 1, \dots, k + T(k)\} \\ \text{Label.Index}_{\{1,0\}} := \{k | 1 \leq k \leq N, \quad T(k) = 0, \quad \text{Nutrition.carbohydrate}(k) \neq 0\} \\ \text{Label.Index}_{\{0,0\}} := \{k | 1 \leq k \leq N, \quad T(k) = 0, \quad \text{Nutrition.carbohydrate}(k) = 0\} \end{cases} \quad (11)$$

The label indexes defined by (11) corresponds to classes “Meal and Exercise”, “no Meal but Exercise”, “no Exercise but Meal”, “neither Meal nor Exercise”, respectively.

Four different configurations of the RNN models were studied to assess the accuracy and performance of each in estimating the joint probability of the carbohydrate intake and physical activity. All four models used 24 past samples of the selected feature variables, and event estimations were performed one sample backward. Estimating the co-occurrences of the external disturbances should be performed at least one step backward as the effect of disturbance variables needs to be seen first, before parameter adjustment and event prediction can be made.

Since the imputation of gaps with a high number of consecutive missing values adversely affects the prediction of meal–exercise classes, all remaining samples with missing values after the data imputation step were excluded from parameter optimization. Excluding missing values inside the input tensor can be carried out either by using a placeholder for missing samples and filtering samples through masking layer or by manually removing incomplete samples.

Each recurrent NN models used in this study encompasses a type of LSTM units [50] (see Figure 2) to capture the time-dependent patterns in the data. The first NN model consists of a masking layer to filter out unimputed samples, followed by a LSTM layer, two dense layers, and a softmax layer to estimate the probability of each class. The LSTM and dense layers undergo training with dropout and parameter regularization strategies to avoid the drastic growth of hyperparameters. Additionally, the recurrent information stream in the LSTM layer was randomly ignored in the calculation at each run. At each layer of the network, the magnitude of both weights and intercept coefficients was penalized by adding a L_1 regularizer term to the loss function. The rectified linear unit (ReLU) activation function was chosen as a nonlinear component in all layers. The input variables of the regular LSTM network will have the shape of $N \times m \times L$, which denotes the size of samples, the size of lagged samples, and the number of feature variables, respectively.

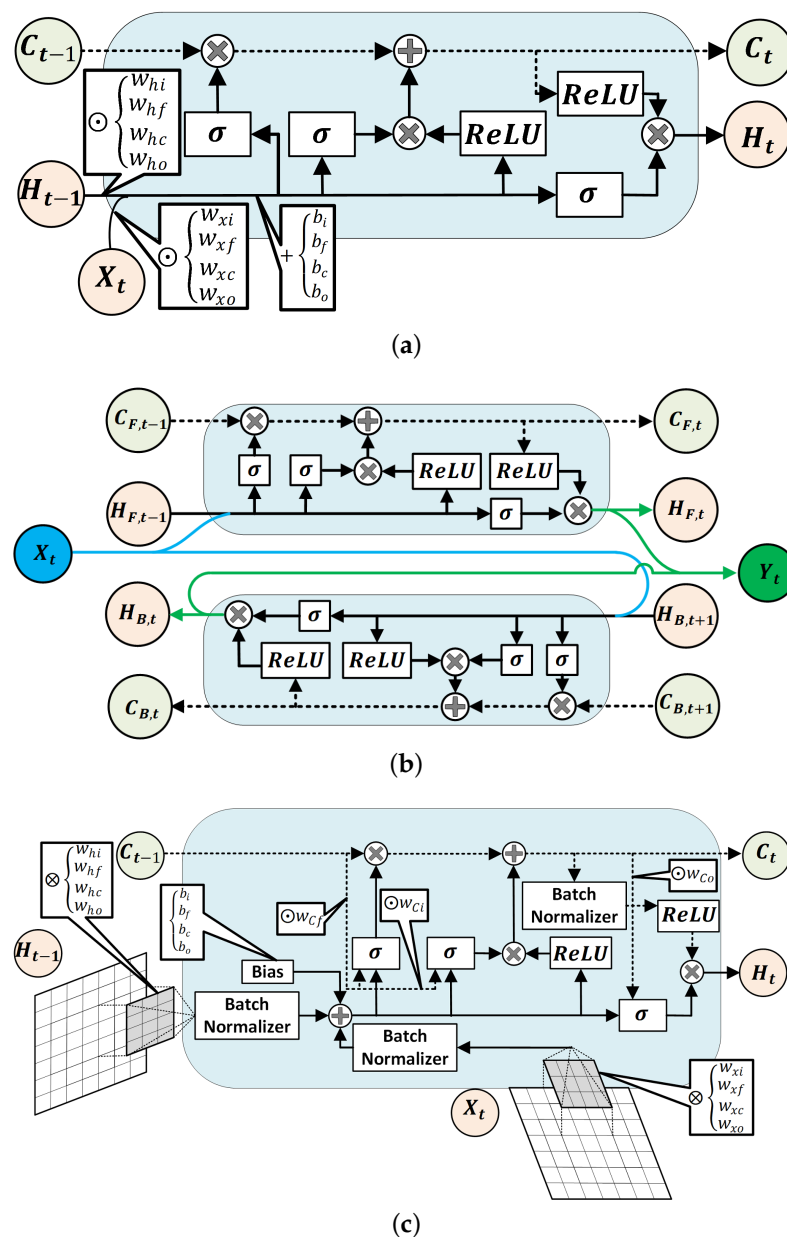


Figure 2. Structures of a regular LSTM unit (a), a Bi-LSTM unit (b), and schematic demonstration of a 2D ConvLSTM cell (c) [50].

The second model encompasses a series of two 1D convolution layers, each one followed by a max pool layer for downsampling feature maps. The output of the second max pool layer was flattened to achieve a time-series extracted feature to feed to the LSTM layer. A dense layer after LSTM was added to the model and the joint probability of events was estimated by calculating the output of the softmax layer. Like the first RNN model, the ReLU activation function was employed in all layers to capture the nonlinearity in the data. A L_1 regularization method was applied to all hyperparameters of the model. Adding convolution layers with repeated operations to an RNN model paves the way for extracting features for the sequence regression or classification problem. This approach has shown a breakthrough in visual time-series prediction from the sequence of images or videos for various problems, such as activity recognition, textual description, and audio and word sequence prediction [51,52]. Time-distributed convolution layers scan and elicit features from each block of the sequence of the data [53]. Therefore, each sample was reshaped into $m \times n \times L$, with $n = 1$ blocks at each sample.

The third classifier has a 2D convolutional LSTM (ConvLSTM) layer, one dropout layer, two dense layers, and a softmax layer for the probability estimation of each class from the sequences of data. A two-dimensional ConvLSTM structure was designed to capture both temporal and spatial correlation in the data, moving pictures in particular, by employing a convolution operation in both input-to-state and state-to-state transitions [50]. In comparison to a regular LSTM cell, ConvLSTMs perform the convolution operation by an internal multiplication of inputs and hidden states into kernel filter matrices (Figure 2c). Similar to previously discussed models, the L_1 regularization constraint and ReLU activation function were considered in constructing the ConvLSTM model. A two-dimensional ConvLSTM import sample of spatiotemporal data in the format of $m \times s \times n \times L$, where $s = 1$ and $n = 1$, stands for the size of the rows and columns of each tensor, and $L = 20$ is the number of channels/features on the data [54].

Finally, the last model comprises two 1D convolution layers, two max pooling layers, a flatten layer, a bidirectional LSTM (Bi-LSTM) layer, a dense layer, and a soft max layer to predict classes. Bi-LSTM units capture the dependency in the sequence of the data in two directions. Hence, as a comparison to a regular LSTM memory unit, Bi-LSTM requires reversely duplicating the same LSTM unit and employing a merging strategy to calculate the output of the cell [55]. The use of this approach was primarily observed in speech recognition tasks, where, instead of real-time interpretation, the whole sequence of the data was analyzed and its superior performance over the regular LSTM was justified [56]. The joint estimation of glycemic events was made one step backward. Therefore, the whole sequence of features were recorded first, and the use of an RNN model with Bi-LSTM units for the detection of unannounced disturbances was quite justifiable. The tensor of input data is similar to LSTM with 1D convolutional layers. Figure 2 is the schematic diagram of a regular LSTM, a Bi-LSTM, and a ConvLSTM unit.

Figure 3 depicts the structure of the four RNN models to estimate the probability of meal consumption, physical activity, and their concurrent occurrence. The main difference between models (a) and (b) in Figure 3 is the convolution and max-pooling layers added before the LSTM layer to extract features map from time series data. Although adding convolutional blocks to an RNN model increases the number of learnable parameters, including weights, biases, and kernel filters, calculating temporal feature maps from input data better discriminates the target classes.

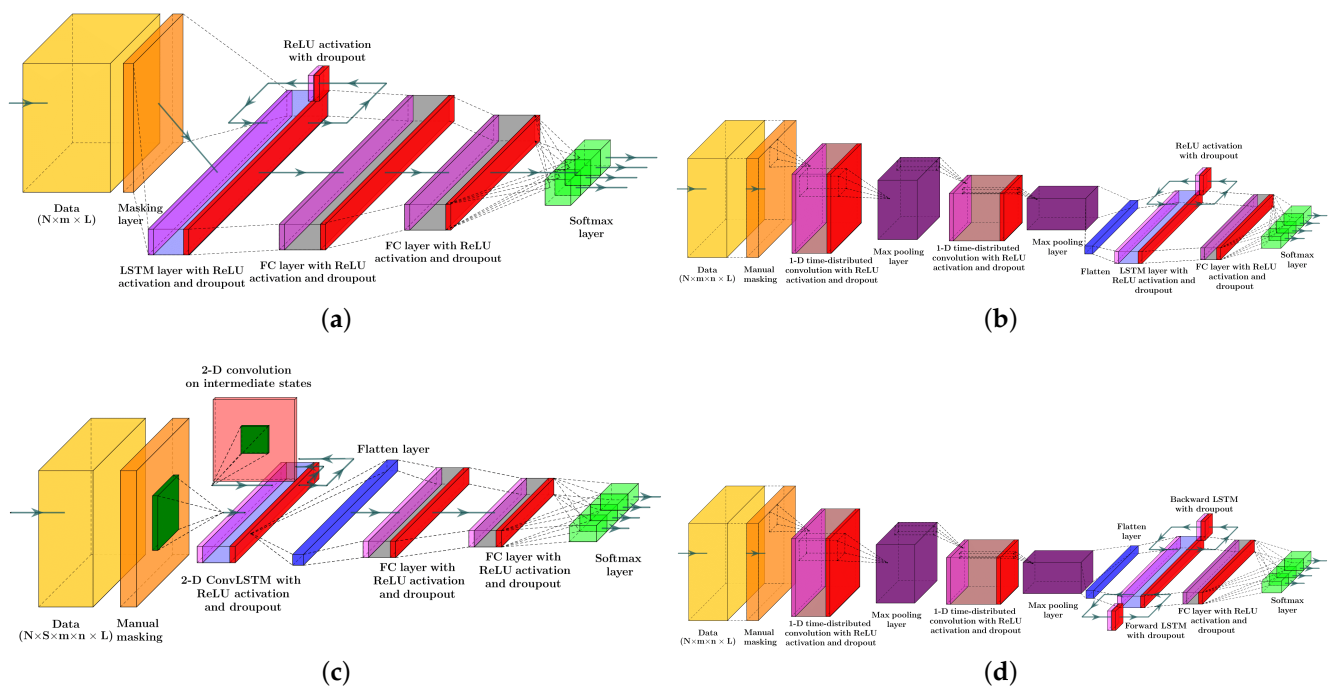


Figure 3. Systematic structures of the different RNN models included in the study: (a) LSTM NN model, (b) LSTM with 1D convolutional layers, (c) 2D ConvLSTM NN model, and (d) Bi-LSTM with 1D convolutional layers. Color dictionary: Yellow: tensor of data, Orange: Masking to exclude missing samples, Magenta: Relu activation, Light blue: LSTM layer, Red: dropout, Grey: dense layer, Green: softmax activation, Blue: flatten layer, Purple: max pool layer, Dark green: kernel filter, Light red: the matrix of intermediate states.

5. Case Study

Eleven datasets containing CGM-sensor-insulin-pump, physical activity, and carbohydrate intake information were selected randomly from subject records for a case study. Data imputation and reconciliation, RNN training, and an evaluation of the results were conducted individually for each subject. Hence, the RNN models were personalized, using only that person's data. All datasets were preprocessed by the procedure elaborated on in the data preprocessing section and feature variables were rescaled to have zero-mean and unit variance. Stratified six-fold cross-validation was applied to 87.5% of samples of each dataset to reduce the variance of predictions. Weight values proportional to the inversion of class sizes were assigned to the corresponding samples to avoid biased predictions caused by imbalanced samples in each class. In order to better assess the performance of each model and to avoid the effects of randomization in the initialization step of the back propagation algorithm, each model was trained five times with different random seeds. Hyperparameters of all models were obtained through an adaptive moment estimation (Adam) optimization algorithm, and 2% of the training sample size was chosen as the size of the training batches. In model training with different random seeds, the number of adjustable parameters, including weights, biases, the size and number of filter kernels, and the learning rate remained constant.

One difficulty associated with convolution layers in models (b) and (d) is the optimization of the hyperparameters of the convolutional layers. Usually, RNN models with convolution layers require a relatively high computation time. As a solution, learning rates with small values are preferred for networks with convolutional layers since they lead to a more optimal solution compared to large learning weights, which may result in non-optimality and instability.

The data preprocessing part of the work was conducted in a Matlab 2019a environment, and Keras/Keras-gpu 2.3.1 were used to construct and train all RNN models. Keras is a high-class API library with Tensorflow as the backend; all are available in the Python environment. We used two computational resources for data preparations and model training. Table 5 provides the details of hardware resources.

Table 5. Hardware specifications.

	Data Preprocessing Task	Model Training Task
CPU Model	Intel i9 9900 k	Intel i7 8700 k
CPU Frequency	3.6–5.0 GHz	3.7–4.7 GHz
Threads	16	12
RAM Capacity	64 GB (DDR IV)	32 GB (DDR IV)
Graphics Processor	RTX 2080 Ti × 2	GTX 1050 Ti (GDDR5)
Graphics Memory	11 GB	4 GB
Clock Frequency	1545–1750 MHz	1290–1392 MHz
Cuda Kernels	4352	768

6. Discussion of Results

Each classifier was evaluated by testing a 12.5% split of all sensor and insulin pump recordings for each subject, corresponding to 3–12 weeks of data for a subject. The average and the standard deviation of performance indexes are reported in Table 6. The lowest performance indexes were achieved by 2D ConvLSTM models. Bi-LSTM with 1D convolution layer RNN models achieve the highest accuracy for six subjects out of eleven, and LSTM with 1D convolution RNN for three subjects. Bi-LSTM with 1D convolution layer RNN models outperformed other models for four subjects, with weighted F1 scores ranging from 91.41–96.26%. Similarly, LSTM models with 1D convolution layers achieved the highest weighted F1 score for another four subjects, with score values within 93.65–96.06%. Glycemic events for the rest of the three subjects showed to be better predicted by regular LSTM models, with a weighted F1 score between 93.31–95.18%. This indicates that 1D convolution improves both the accuracy and F1 scores for most of the subjects. Based on the number of adjustable parameters for the four different RNN models used for a specific subject, LSTMs are the most computational demanding blocks in the model. To assess the computational load of developing the various RNN models, we compared the number of learnable parameters (details provided in Supplementary Materials). These values can be highly informative, as the number of dropouts in each model and the number of learnable parameters at each epoch (iteration) are invariant.

A comparison between 1D conv-LSTM and 1D-Bi-LSTM for one randomly selected subject shows that the number of learnable parameters increases by at least 54%, mainly stemming from an extra embedded LSTM in the bidirectional layer (Table S1). While comparing adjustable parameters may not be the most accurate way of determining the computational loads for training the models, they provide a good reference to compare the computational burden of different RNN models.

Figure 4 displays a random day selected from the test data to compare the effectiveness of each RNN model in detecting meal and exercise disturbances. Among four possible realizations for the occurrence of events, detecting joint events, $Class_{1,1}$, is more challenging as it usually shows overlaps with $Class_{0,1}$ and $Class_{1,0}$. Another reason for the lower detection is the lack of enough information on $Class_{1,1}$, knowing that people would usually rather have a small snack before and after exercise sessions over having a rescue carbohydrate during physical activity. Furthermore, the AID systems used by subjects automatically record only CGM and insulin infusion values, and meal and physical activity sessions need to be manually entered to the device, which is, at times, an action that may be forgotten by the subject. Meal consumption and physical activity are two prominent disturbances that disrupt BGC regulation, but their opposite effect on BGC makes the prediction of $Class_{1,1}$ less critical than each of meal intake or only physical activity classes.

Table 6. The average performance indexes of LSTM, LSTM with 1D convolution layers, 2D ConvLSTM, and Bi-LSTM with 1D convolution layers RNN models for the event detection problem. Standard deviations are given in parentheses and values with bold notation denote the highest performance indexes.

Subject No./Model	Total Accuracy (%)	Weighted Recall (%)	Weighted Precision (%)	Weighted F1 Score (%)
1 {	LSTM	92.03 (0.37)	92.03 (0.37)	94.59 (0.29)
	LSTM(1D Convolution)	94.61 (0.15)	94.61 (0.15)	97.67 (0.20)
	2D ConvLSTM	89.89 (0.27)	89.89 (0.27)	94.36 (0.16)
	Bi – LSTM(1D Convolution)	94.72 (0.31)	94.72 (0.31)	96.85 (0.34)
2 {	LSTM	93.17 (0.21)	93.17 (0.21)	96.17 (0.09)
	LSTM(1D Convolution)	94.69 (0.33)	94.69 (0.33)	96.58 (0.20)
	2D ConvLSTM	91.29 (0.38)	91.29 (0.38)	95.90 (0.20)
	Bi – LSTM(1D Convolution)	93.56 (0.21)	93.56 (0.21)	96.99 (0.25)
3 {	LSTM	89.93 (0.22)	89.93 (0.22)	93.98 (0.24)
	LSTM(1D Convolution)	88.98 (0.28)	88.98 (0.28)	93.53 (0.09)
	2D ConvLSTM	88.40 (0.25)	88.40 (0.25)	93.13 (0.12)
	Bi – LSTM(1D Convolution)	89.98 (0.17)	89.98 (0.17)	93.87 (0.05)
4 {	LSTM	92.55 (0.27)	92.55 (0.27)	95.49 (0.09)
	LSTM(1D Convolution)	94.67 (0.31)	94.67 (0.31)	96.62 (0.17)
	2D ConvLSTM	88.87 (0.33)	88.87 (0.33)	94.61 (0.16)
	Bi – LSTM(1D Convolution)	94.49 (0.28)	94.49 (0.28)	96.47 (0.25)
5 {	LSTM	94.41 (0.21)	94.41 (0.21)	96.46 (0.10)
	LSTM(1D Convolution)	91.65 (0.26)	91.65 (0.26)	96.16 (0.12)
	2D ConvLSTM	89.81 (0.18)	89.81 (0.18)	95.62 (0.10)
	Bi – LSTM(1D Convolution)	92.66 (0.21)	92.66 (0.21)	96.93 (0.28)
6 {	LSTM	94.50 (0.27)	94.50 (0.27)	95.15 (0.22)
	LSTM(1D Convolution)	95.67 (0.26)	95.67 (0.26)	96.69 (0.09)
	2D ConvLSTM	91.10 (0.25)	91.10 (0.25)	94.37 (0.10)
	Bi – LSTM(1D Convolution)	95.86 (0.17)	95.86 (0.17)	96.72 (0.21)
7 {	LSTM	91.81 (0.22)	91.81 (0.22)	95.43 (0.22)
	LSTM(1D Convolution)	89.85 (0.22)	89.85 (0.22)	94.19 (0.17)
	2D ConvLSTM	87.47 (0.27)	87.47 (0.27)	99.83 (0.22)
	Bi – LSTM(1D Convolution)	90.03 (0.24)	90.03 (0.24)	95.01 (0.16)
8 {	LSTM	89.19 (0.22)	89.19 (0.22)	99.68(0.27)
	LSTM(1D Convolution)	90.99 (0.18)	90.99 (0.18)	97.68 (0.19)
	2D ConvLSTM	83.92 (0.28)	83.92 (0.28)	94.60 (0.33)
	Bi – LSTM(1D Convolution)	91.42 (0.30)	91.42 (0.30)	95.17 (0.37)
9 {	LSTM	92.70 (0.18)	92.70 (0.18)	96.53 (0.28)
	LSTM(1D Convolution)	93.10 (0.32)	93.10 (0.32)	95.70 (0.31)
	2D ConvLSTM	91.56 (0.34)	91.56 (0.34)	94.81 (0.30)
	Bi – LSTM(1D Convolution)	93.33 (0.27)	93.33 (0.27)	97.24 (0.21)
10 {	LSTM	89.30 (0.34)	89.30 (0.34)	95.57 (0.08)
	LSTM(1D Convolution)	91.89 (0.25)	91.89 (0.25)	96.14 (0.08)
	2D ConvLSTM	85.87 (0.27)	85.87 (0.27)	95.69 (0.08)
	Bi – LSTM(1D Convolution)	89.51 (0.25)	89.51 (0.25)	96.11 (0.03)
11 {	LSTM	87.17 (0.44)	87.17 (0.44)	92.26 (0.12)
	LSTM(1D Convolution)	89.41 (0.35)	89.41 (0.35)	94.39 (0.06)
	2D ConvLSTM	86.90 (0.29)	86.90 (0.29)	94.84 (0.21)
	Bi – LSTM(1D Convolution)	89.62 (0.15)	89.62 (0.15)	96.51 (0.13)

The confusion matrices of the classification results for one of the subjects (No. 2) are summarized in Table 7. As can be observed from Figure 4 and Table 7, detecting $Class_{0,1}$ (physical activity) is more challenging in comparison to the carbohydrate intake ($Class_{1,0}$) and $Class_{0,0}$ (no meal or exercise). One reason for this difficulty is the lack of biosignal information, such as 3D accelerometer, blood volume pulse, and heart rate data. Some erroneous detections, such as confusing meals and exercise, are dangerous, since meals necessitate an insulin bolus while exercise lowers BGC, and the elimination of insulin

infusion and/or increase in target BGC are needed. RNNs with LSTM and 1D convolution layers provide the best overall performance in minimizing such confusions: two meals events are classified as exercise (0.003%) and eight exercise events are classified as meals (0.125%).

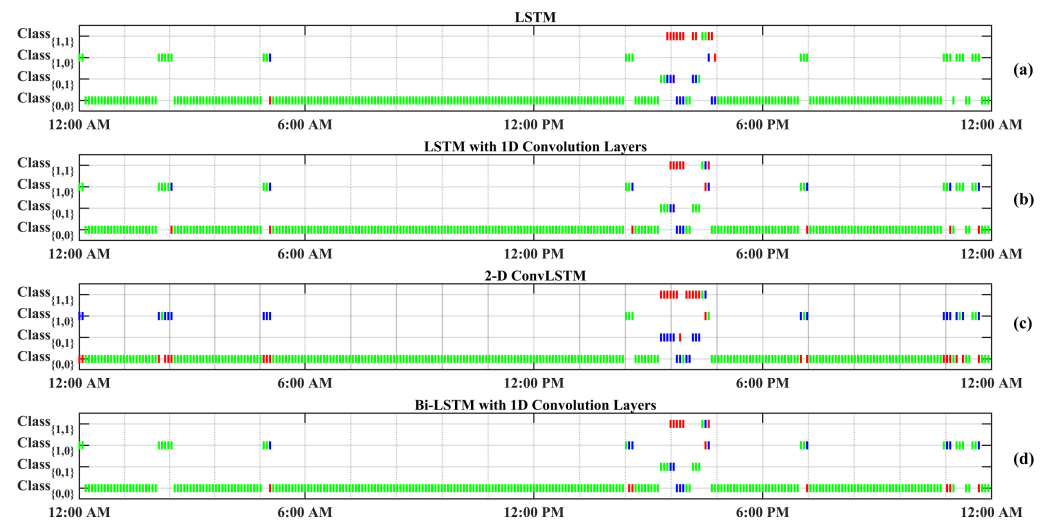


Figure 4. One-step-backward predicted Meal and Exercise events for one randomly selected dataset (Subject 2). Vertical green bars represent correctly predicted classes. Vertical red bars denote incorrectly predicted classes, and their actual labels are shown by blue bars. Class Dictionary: $Class_{0,0}$: “neither Meal nor Exercise”, $Class_{0,1}$: “only Exercise”, $Class_{1,0}$: “only Meal”, $Class_{1,1}$: “Meal and Exercise”.

Two limitations of the study are the quality and accuracy of data collected in free living and the variables that are measured. As stated in the Introduction and Data Preprocessing sections, the missing data in the time series of CGM readings is one limitation that we addressed by developing data preprocessing techniques. The second limitation is the number of variables that are measured. In this data set, there are only CGM and insulin pump data and the voluntary information provided by the patients about meal consumption and exercising. This information is usually incomplete (sometimes people may forget or have no time to enter this information). These events can be captured objectively by other measurements from wearable devices. Such data were not available in this data set and limited the accuracy of the results, especially when the meal and exercise occurred concurrently.

The proportion of correctly detected exercise and meal events to all actual exercise and meal events for all subjects reveals that a series of convolution–max-pooling layers could elicit informative feature maps for classification efficiently. Although augmented features, such as the first and second derivatives of CGM and PIC, enhance the prediction power of the NN models, the secondary feature maps, extracted from all primary features, show to be a better fit for this classification problem. In addition, repeated 1D kernel filters in convolution layers better suit the time-series nature of the data, as opposed to extracting feature maps by utilizing 2D convolution filters on the data.

Table 7. Confusion matrices calculated from the predicted and actual classes of testing samples collected from Subject 2.

Actual						Actual					
		Class _{0,0}	Class _{0,1}	Class _{1,0}	Class _{1,1}			Class _{0,0}	Class _{0,1}	Class _{1,0}	Class _{1,1}
Predicted	Class _{0,0}	11,154	12	46	2	Predicted	Class _{0,0}	11,297	14	8	1
	Class _{0,1}	310	598	2	0		Class _{0,1}	274	596	2	2
	Class _{1,0}	263	21	593	1		Class _{1,0}	311	8	655	1
	Class _{1,1}	217	5	27	6		Class _{1,1}	62	18	3	5
(a) LSTM						(b) LSTM (1D Convolution)					
Actual						Actual					
		Class _{0,0}	Class _{0,1}	Class _{1,0}	Class _{1,1}			Class _{0,0}	Class _{0,1}	Class _{1,0}	Class _{1,1}
Predicted	Class _{0,0}	10,984	14	29	1	Predicted	Class _{0,0}	11,324	14	29	1
	Class _{0,1}	560	574	16	2		Class _{0,1}	110	537	16	2
	Class _{1,0}	157	6	541	2		Class _{1,0}	257	6	538	2
	Class _{1,1}	243	42	82	4		Class _{1,1}	253	79	85	4
(c) 2D ConvLSTM						(d) Bi-LSTM (1D Convolution)					

7. Conclusions

This work focuses on developing RNN models for detection and classification tasks using time series data containing missing and erroneous values. The first modeling issue arose from the quality of the recorded data in free living. An outlier rejection algorithm was developed based on multivariable statistical analysis and signal denoising by decomposition of the Hankel matrix of CGM recordings. A multivariate approach based on PPCA for CGM sample imputation was used to keep the harmony and relationship among the variables. The second issue addressed is the detection of events that affect the behavior of dynamic systems and the classification of these events. Four different RNN models were developed to detect meal and exercise events in the daily lives of individuals with T1D. The results indicate that models with 1D convolution layers can classify events better than regular LSTM RNN and 2D ConvLSTM RNN models, with very low confusion between the events that may cause dangerous situations by prompting erroneous interventions, such as giving insulin boluses during exercise.

Supplementary Materials: The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/biomedinformatics2020019/s1>, File S1: Assessing the computational load of training the RNN models.

Author Contributions: M.R.A., M.R., X.S., M.S., A.S., K.K. and A.C. conceived the research. M.R.A. and M.R. developed the theory, and M.R.A. performed the computations. M.R.A., M.R. and A.C. wrote the manuscript. All authors discussed the results and contributed to the final manuscript. A.C. supervised the project. All authors have read and agreed to the published version of the manuscript.

Funding: Financial support from JDRF under Grant No. 1-SRA-2019-S-B, NIH under Grant No. K25 HL141634 and the Hyosung S. R. Cho Endowed Chair to Ali Cinar at Illinois Institute of Technology are gratefully acknowledged.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cai, L.; Zhu, Y. The challenges of data quality and data quality assessment in the big data era. *Data Sci. J.* **2015**, *14*, 2. [CrossRef]
2. Centers for Disease Control and Prevention. *National Diabetes Statistics Report, 2020*; Centers for Disease Control and Prevention, US Department of Health and Human Services: Atlanta, GA, USA, 2020.
3. Roglic, G. WHO Global report on diabetes: A summary. *Int. J. Noncommun. Dis.* **2016**, *1*, 3. [CrossRef]
4. Horowitz, M.E.; Kaye, W.A.; Pepper, G.M.; Reynolds, K.E.; Patel, S.R.; Knudson, K.C.; Kale, G.K.; Gutierrez, M.E.; Cotto, L.A.; Horowitz, B.S. An analysis of Medtronic MiniMed 670G insulin pump use in clinical practice and the impact on glycemic control, quality of life, and compliance. *Diabetes Res. Clin. Pract.* **2021**, *177*, 108876. [CrossRef] [PubMed]
5. Berget, C.; Lange, S.; Messer, L.; Forlenza, G.P. A clinical review of the t:slim X2 insulin pump. *Expert Opin. Drug Deliv.* **2020**, *17*, 1675–1687. [CrossRef]
6. Cobry, E.C.; Berget, C.; Messer, L.H.; Forlenza, G.P. Review of the Omnipod® 5 Automated Glucose Control System Powered by Horizon™ for the treatment of Type 1 diabetes. *Ther. Deliv.* **2020**, *11*, 507–519. [CrossRef]
7. Sangave, N.A.; Aungst, T.D.; Patel, D.K. Smart Connected Insulin Pens, Caps, and Attachments: A Review of the Future of Diabetes Technology. *Diabetes Spectr.* **2019**, *32*, 378–384. [CrossRef]
8. Hoskins, M. New Diabetes Technology Expected in 2022. Available online: <https://www.healthline.com/diabetesmine/new-diabetes-technology-in-2022> (accessed on 20 May 2022).
9. Tanzi, M.G. FDA approves first interchangeable biosimilar insulin product for treatment of diabetes. *Pharmacy Today* **2021**, *27*, 21.
10. Sevil, M.; Rashid, M.; Hajizadeh, I.; Askari, M.R.; Hobbs, N.; Brandt, R.; Park, M.; Quinn, L.; Cinar, A. Automated insulin delivery systems for people with type 1 diabetes. In *Drug Delivery Devices and Therapeutic Systems*; Chappel, E., Ed.; Developments in Biomedical Engineering and Bioelectronics, Academic Press: Cambridge, MA, USA, 2021; Chapter 9; pp. 181–198.
11. Boughton, C.K.; Hovorka, R. New closed-loop insulin systems. *Diabetologia* **2021**, *64*, 1007–1015. [CrossRef]
12. Brown, S.A.; Kovatchev, B.P.; Raghinaru, D.; Lum, J.W.; Buckingham, B.A.; Kudva, Y.C.; Laffel, L.M.; Levy, C.J.; Pinsky, J.E.; Wadwa, R.P.; et al. Six-Month Randomized, Multicenter Trial of Closed-Loop Control in Type 1 Diabetes. *N. Engl. J. Med.* **2019**, *381*, 1707–1717. [CrossRef]
13. Forlenza, G.P.; Buckingham, B.A.; Brown, S.A.; Bode, B.W.; Levy, C.J.; Criego, A.B.; Wadwa, R.P.; Cobry, E.C.; Slover, R.J.; Messer, L.H.; et al. First Outpatient Evaluation of a Tubeless Automated Insulin Delivery System with Customizable Glucose Targets in Children and Adults with Type 1 Diabetes. *Diabetes Technol. Ther.* **2021**, *23*, 410–424. [CrossRef]
14. Ware, J.; Hovorka, R. Recent advances in closed-loop insulin delivery. *Metab.-Clin. Exp.* **2022**, *127*, 154953. [CrossRef] [PubMed]
15. Garcia-Tirado, J.; Lv, D.; Corbett, J.P.; Colmegna, P.; Breton, M.D. Advanced hybrid artificial pancreas system improves on unannounced meal response—In silico comparison to currently available system. *Comput. Methods Programs Biomed.* **2021**, *211*, 106401. [CrossRef] [PubMed]
16. Haidar, A.; Legault, L.; Raffray, M.; Gouchie-Provencher, N.; Jacobs, P.G.; El-Fathi, A.; Rutkowski, J.; Messier, V.; Rabasa-Lhoret, R. Comparison Between Closed-Loop Insulin Delivery System (the Artificial Pancreas) and Sensor-Augmented Pump Therapy: A Randomized-Controlled Crossover Trial. *Diabetes Technol. Ther.* **2021**, *23*, 168–174. [CrossRef]
17. Paldus, B.; Lee, M.H.; Morrison, D.; Zaharieva, D.P.; Jones, H.; Obeyesekere, V.; Lu, J.; Vogrin, S.; LaGerche, A.; McAuley, S.A.; et al. First Randomized Controlled Trial of Hybrid Closed Loop Versus Multiple Daily Injections or Insulin Pump Using Self-Monitoring of Blood Glucose in Free-Living Adults with Type 1 Diabetes Undertaking Exercise. *J. Diabetes Sci. Technol.* **2021**, *15*, 1399–1401. [CrossRef]
18. Ekhlaspour, L.; Forlenza, G.P.; Chernavsky, D.; Maahs, D.M.; Wadwa, R.P.; Deboer, M.D.; Messer, L.H.; Town, M.; Pinnata, J.; Kruse, G.; et al. Closed loop control in adolescents and children during winter sports: Use of the Tandem Control-IQ AP system. *Pediatr. Diabetes* **2019**, *20*, 759–768. [CrossRef]
19. Deshpande, S.; Pinsky, J.E.; Church, M.M.; Piper, M.; Andre, C.; Massa, J.; Doyle, F.J., III; Eisenberg, D.M.; Dassau, E. Randomized Crossover Comparison of Automated Insulin Delivery Versus Conventional Therapy Using an Unlocked Smartphone with Scheduled Pasta and Rice Meal Challenges in the Outpatient Setting. *Diabetes Technol. Ther.* **2020**, *22*, 865–874. [CrossRef] [PubMed]
20. Wilson, L.M.; Jacobs, P.G.; Riddell, M.C.; Zaharieva, D.P.; Castle, J.R. Opportunities and challenges in closed-loop systems in type 1 diabetes. *Lancet Diabetes Endocrinol.* **2022**, *10*, 6–8. [CrossRef]
21. Franc, S.; Benhamou, P.Y.; Borot, S.; Chaillous, L.; Delemer, B.; Doron, M.; Guerci, B.; Hanaire, H.; Hunecker, E.; Jeandidier, N.; et al. No more hypoglycaemia on days with physical activity and unrestricted diet when using a closed-loop system for 12 weeks: A post hoc secondary analysis of the multicentre, randomized controlled Diabeloop WP7 trial. *Diabetes Obes. Metab.* **2021**, *23*, 2170–2176. [CrossRef]
22. Jeyaventhana, R.; Gallen, G.; Choudhary, P.; Hussain, S. A real-world study of user characteristics, safety and efficacy of open-source closed-loop systems and Medtronic 670G. *Diabetes Obes. Metab.* **2021**, *23*, 1989–1994. [CrossRef]
23. Jennings, P.; Hussain, S. Do-it-yourself artificial pancreas systems: A review of the emerging evidence and insights for healthcare professionals. *J. Diabetes Sci. Technol.* **2020**, *14*, 868–877. [CrossRef]
24. Hajizadeh, I.; Askari, M.R.; Kumar, R.; Zavala, V.M.; Cinar, A. Integrating MPC with Learning-Based and Adaptive Methods to Enhance Safety, Performance and Reliability in Automated Insulin Delivery. *IFAC-PapersOnLine* **2020**, *53*, 16149–16154. [CrossRef]

25. Hajizadeh, I.; Askari, M.R.; Sevil, M.; Hobbs, N.; Brandt, R.; Rashid, M.; Cinar, A. Adaptive control of artificial pancreas systems for treatment of type 1 diabetes. In *Control Theory in Biomedical Engineering*; Boubaker, O., Ed.; Academic Press: Cambridge, MA, USA, 2020; Chapter 3; pp. 63–81.
26. Askari, M.R.; Hajizadeh, I.; Rashid, M.; Hobbs, N.; Zavala, V.M.; Cinar, A. Adaptive-learning model predictive control for complex physiological systems: Automated insulin delivery in diabetes. *Annu. Rev. Control.* **2020**, *50*, 1–12. [[CrossRef](#)]
27. Garcia-Tirado, J.; Brown, S.A.; Laichuthai, N.; Colmegna, P.; Koravi, C.L.; Ozaslan, B.; Corbett, J.P.; Barnett, C.L.; Pajewski, M.; Oliveri, M.C.; et al. Anticipation of Historical Exercise Patterns by a Novel Artificial Pancreas System Reduces Hypoglycemia During and After Moderate-Intensity Physical Activity in People with Type 1 Diabetes. *Diabetes Technol. Ther.* **2021**, *23*, 277–285. [[CrossRef](#)] [[PubMed](#)]
28. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)]
29. Yu, Y.; Si, X.; Hu, C.; Zhang, J. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural Comput.* **2019**, *31*, 1235–1270. [[CrossRef](#)]
30. Efron, B. Missing data, imputation, and the bootstrap. *J. Am. Stat. Assoc.* **1994**, *89*, 463–475. [[CrossRef](#)]
31. Moritz, S.; Bartz-Beielstein, T. imputeTS: Time series missing value imputation in R. *R J.* **2017**, *9*, 207. [[CrossRef](#)]
32. Ahola, A.J.; Muttter, S.; Forsblom, C.; Harjutsalo, V.; Groop, P.H. Meal timing, meal frequency, and breakfast skipping in adult individuals with type 1 diabetes—associations with glycaemic control. *Sci. Rep.* **2019**, *9*, 1–10. [[CrossRef](#)]
33. Tipping, M.E.; Bishop, C.M. Probabilistic principal component analysis. *J. R. Stat. Soc. Ser. Stat. Methodol.* **1999**, *61*, 611–622. [[CrossRef](#)]
34. Ilin, A.; Raiko, T. Practical approaches to principal component analysis in the presence of missing values. *J. Mach. Learn. Res.* **2010**, *11*, 1957–2000.
35. Hoaglin, D.C.; John, W. Tukey and Data Analysis. *Stat. Sci.* **2003**, *18*, 311–318. [[CrossRef](#)]
36. Bakshi, B.; Stephanopoulos, G. Representation of process trends-III. Multiscale extraction of trends from process data. *Comput. Chem. Eng.* **1994**, *18*, 267–302. [[CrossRef](#)]
37. Cheung, J.Y.; Stephanopoulos, G. Representation of process trends-Part I. A formal representation framework. *Comput. Chem. Eng.* **1990**, *14*, 495–510. [[CrossRef](#)]
38. Samadi, S.; Turksoy, K.; Hajizadeh, I.; Feng, J.; Sevil, M.; Cinar, A. Meal Detection and Carbohydrate Estimation Using Continuous Glucose Sensor Data. *IEEE J. Biomed. Health Inf.* **2017**, *21*, 619–627. [[CrossRef](#)] [[PubMed](#)]
39. Samadi, S.; Rashid, M.; Turksoy, K.; Feng, J.; Hajizadeh, I.; Hobbs, N.; Lazaro, C.; Sevil, M.; Littlejohn, E.; Cinar, A. Automatic Detection and Estimation of Unannounced Meals for Multivariable Artificial Pancreas System. *Diabetes Technol. Ther.* **2018**, *20*, 235–246. [[CrossRef](#)] [[PubMed](#)]
40. Eberle, C.; Ament, C. The Unscented Kalman Filter estimates the plasma insulin from glucose measurement. *Biosystems* **2011**, *103*, 67–72. [[CrossRef](#)] [[PubMed](#)]
41. Hajizadeh, I.; Rashid, M.; Turksoy, K.; Samadi, S.; Feng, J.; Frantz, N.; Sevil, M.; Cengiz, E.; Cinar, A. Plasma insulin estimation in people with type 1 diabetes mellitus. *Ind. Eng. Chem. Res.* **2017**, *56*, 9846–9857. [[CrossRef](#)]
42. Hajizadeh, I.; Rashid, M.; Samadi, S.; Feng, J.; Sevil, M.; Hobbs, N.; Lazaro, C.; Maloney, Z.; Brandt, R.; Yu, X.; et al. Adaptive and Personalized Plasma Insulin Concentration Estimation for Artificial Pancreas Systems. *J. Diabetes Sci. Technol.* **2018**, *12*, 639–649. [[CrossRef](#)]
43. Hovorka, R.; Canonico, V.; Chassin, L.J.; Haueter, U.; Massi-Benedetti, M.; Federici, M.O.; Pieber, T.R.; Schaller, H.C.; Schaupp, L.; Vering, T.; et al. Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes. *Physiol. Meas.* **2004**, *25*, 905–920. [[CrossRef](#)]
44. Ivers, N.M.; Jiang, M.; Alloo, J.; Singer, A.; Ngui, D.; Casey, C.G.; Catherine, H.Y. Diabetes Canada 2018 clinical practice guidelines: Key messages for family physicians caring for patients living with type 2 diabetes. *Can. Fam. Physician* **2019**, *65*, 14–24.
45. Care, F. Standards of Medical Care in Diabetes 2019. *Diabetes Care* **2019**, *42*, S124–S138.
46. NICE. *Type 1 Diabetes in Adults: Diagnosis and Management*; National Institute for Health and Care Excellence (NICE): London, UK, 2015; pp. 1–87.
47. Arulampalam, M.; Maskell, S.; Gordon, N.; Clapp, T. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Process.* **2002**, *50*, 174–188. [[CrossRef](#)]
48. Pitt, M.; Silva, R.; Giordani, P.; Kohn, R. Auxiliary particle filtering within adaptive Metropolis-Hastings sampling. *arXiv* **2010**, arXiv:1006.1914.
49. Pudil, P.; Novovičová, J.; Kittler, J. Floating search methods in feature selection. *Pattern Recognit. Lett.* **1994**, *15*, 1119–1125. [[CrossRef](#)]
50. Shi, X.; Chen, Z.; Wang, H.; Yeung, D.Y.; Wong, W.K.; Woo, W.c. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 802–810.
51. Vinyals, O.; Toshev, A.; Bengio, S.; Erhan, D. Show and tell: A neural image caption generator. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3156–3164.
52. Donahue, J.; Anne Hendricks, L.; Guadarrama, S.; Rohrbach, M.; Venugopalan, S.; Saenko, K.; Darrell, T. Long-term recurrent convolutional networks for visual recognition and description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 2625–2634.

-
53. Géron, A. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*; O'Reilly Media: Newton, MA, USA, 2019.
 54. Brownlee, J. *Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python*; Machine Learning Mastery. 2018. Available online: <https://machinelearningmastery.com/deep-learning-for-time-series-forecasting/> (accessed on 20 May 2022).
 55. Schuster, M.; Paliwal, K. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681. [[CrossRef](#)]
 56. Graves, A.; Schmidhuber, J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw.* **2005**, *18*, 602–610. [[CrossRef](#)]