Imperial College London

Deep Learning in Diabetes Management

Taiyu Zhu

Department of Electrical and Electronic Engineering
Imperial College London

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy of Imperial College London
and the Diploma of Imperial College London

September 2022

Statement of Originality

I hereby declare that the content of this thesis is the product of my own work under the supervision of Prof. Pantelis Georgiou and Dr. Kezhi Li. Ideas and information derived from the work of other people have been appropriately acknowledged with the standard referencing practices employed in this discipline.

Taiyu Zhu

Copyright Declaration

Abstract

Diabetes is a group of chronic metabolic disorders that affect almost half a billion people worldwide. Despite the rapid advancement of wearable devices, such as continuous glucose monitoring (CGM), maintaining blood glucose (BG) levels in a therapeutically appropriate range has been a heavy daily burden for people living with type 1 diabetes. Due to the large inter- and intra-subject variability, finding optimal personalised treatment is still an open problem. In this thesis, a wide range of novel deep learning technology is investigated to enhance diabetes management and tackle the challenges in BG prediction, glycaemic control, BG time series generation, and digital health systems.

Combing the latest advances in evidential deep learning and meta-learning, this thesis proposes the Fast-adaptive and Confident Neural Network (FCNN), a novel deep learning framework for personalised BG prediction, which incorporates model confidence and enables fast adaptation to address the challenges in clinical settings. The proposed algorithm was evaluated on three clinical datasets and achieved state-of-the-art performance. Then, the physiological data measured by wearable wristband sensors were integrated into BG prediction using the FCNN framework, which significantly improved the model performance.

Subsequently, deep reinforcement learning (DRL) is explored in glycaemic control. First, a novel algorithm based on double deep Q-learning is proposed to control basal insulin and glucagon delivery in the artificial pancreas for single- or dual-hormone therapy. Then, an actor-critic algorithm is applied to develop a novel insulin advisor to recommend meal insulin bolus. The results of *in silico* trials demonstrated that the proposed DRL control algorithms significantly enhanced the percentage of time spent in the target BG range and reduced hypoglycaemia and hyperglycaemia. Furthermore, a novel offline DRL and off-policy evaluation framework is proposed for basal insulin control, which enables DRL models to be developed in a safe and offline process. The framework was evaluated on both *in silico* and clinical datasets and improved various clinical metrics.

To generate synthetic BG time series for data augmentation, this thesis introduces GluGAN, a novel framework based on generative adversarial networks. By integrating a supervised learning loss into adversarial training, GluGAN captures autoregressive temporal dynamics and generates high-quality synthetic BG data for the three clinical datasets. In the experiments on BG prediction, training data augmented by GluGAN significantly improved the performance of three classic data-driven algorithms.

Finally, this thesis proposes a novel Internet of Medical Things (IoMT) framework for digital health systems in diabetes management. The centre of the proposed framework is an IoMT-enabled wearable wristband that comprises a low-cost and low-power system on a chip to communicate CGM and provide decision support by edge computing. In addition, a smartphone app is designed for data visualisation, while desktop and cloud platforms are proposed for data storage and model training. As a use case, an embedded BG prediction algorithm is developed through the FCNN framework and implemented on the wristband. The optimised hardware design results in extremely low energy consumption for edge inference and wireless connectivity on the wristband. The use of the IoMT framework notably improved glycaemic control in a hardware-in-the-loop *in silico* trial.

Acknowledgment

First and foremost, I would like to express my deepest gratitude to my supervisor Prof. Pantelis Georgiou for granting me the opportunity to conduct this cutting-edge research and giving me endless guidance, support, and encouragement throughout my PhD research. His knowledge and insight have allowed me to explore many innovative ideas and new possibilities in this emerging area. I would like to express my sincere appreciation to Dr. Kezhi Li who has been my co-supervisor since the beginning of my research. He led me to the academic world and has taught me invaluable lessons. Our fruitful discussions have contributed to a series of exciting research in this thesis. I feel privileged to have worked closely with my supervisors and look forward to continuing the collaboration in my future academic career.

My special thanks to Dr. Pau Herrero for sharing his knowledge and providing continuous guidance and support. His constructive suggestions have been extremely helpful for this research. I am grateful to Prof. Nick Oliver and the clinical research team for the insight and support they have provided.

I am also thankful to my examiners, Prof. Krystian Mikolajczyk and Prof. Jorge Bondia, for their valuable feedback and insightful suggestions that considerably improved this thesis.

I would like to acknowledge the President's PhD Scholarship at Imperial College London for funding my research, which has allowed me to stay dedicated to pursuing my research goals. I would also like to thank the Department of Electrical and Electronic Engineering for recommending me to this prestigious award.

I would like to thank my colleagues in the Centre for Bio-Inspired Technology for their help and friendship. My sincere thanks to Dr. Yiwei Wu, who supported, impacted, and inspired me on this incredible journey.

And most of all, I would like to express my warmest thanks to my family, and especially to my parents who always believe in me, support me, and encourage me to follow my dreams.

Contents

List of Figures

List of Tables

List of Publications

The publications arising from this research are listed as follows.

Peer-Reviewed Journal Publications

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

- **T. Zhu**, K. Li, P. Herrero, P. Georgiou, "Personalized blood glucose prediction for type 1 diabetes using evidential deep learning and meta-learning," *IEEE Transactions on Biomedical Engineering*, 2022.

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

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

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

- **T. Zhu**, K. Li, L. Kuang, P. Herrero and P. Georgiou, "An insulin bolus advisor for type 1 diabetes using deep reinforcement learning," *Sensors*, vol. 20, no. 18, p. 5058, 2020. (**Editor's Choice Article**)

- **T. Zhu**, K. Li, J. Chen, P. Herrero, P. Georgiou, "Dilated recurrent neural networks for glucose forecasting in type 1 diabetes," *Journal of Healthcare Informatics Research*, vol. 4, no. 3, pp. 308–324, 2020.

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

Submitted Journals

- **T. Zhu**, K. Li, P. Herrero and P. Georgiou, "GluGAN: Generating Personalized Glucose Time Series Using Generative Adversarial Networks," Submitted to *IEEE Journal of Biomedical and Health Informatics.*

- **T. Zhu**, K. Li, P. Herrero and P. Georgiou, "Offline Deep Reinforcement Learning and Off-Policy Evaluation for Personalized Blood Glucose Control in Type 1 Diabetes," Submitted to *IEEE Journal of Biomedical and Health Informatics.*

- G. Noaro, **T. Zhu**, G. Cappon, A. Facchinetti, P. Georgiou, "A Personalized and Adaptive Insulin Bolus Calculator Based on Double Deep Q-Learning to Improve Type 1 Diabetes Management," Submitted to *IEEE Journal of Biomedical and Health Informatics.*

Peer-Reviewed Conference Proceedings

- **T. Zhu**, L. Kuang, K. Li, J. Zeng, P. Herrero and P. Georgiou, "Blood Glucose Prediction in Type 1 Diabetes Using Deep Learning on the Edge," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2021.

- L. Kuang, **T. Zhu**, K. Li, J. Daniels, P. Herrero, P. Georgiou. "Live demonstration: An IoT wearable device for real-time blood glucose prediction with edge AI," in *IEEE Biomedical Circuits and Systems Conference (BIOCAS)*, 2021.

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

- **T. Zhu**, X. Yao, K. Li, P. Herrero, and P. Georgiou, "Blood glucose prediction for type 1 diabetes using generative adversarial networks," in *the 5th International Workshop on Knowledge Discovery in Healthcare Data, European Conference on Artificial Intelligence*, pp.90-94, 2020.

Conference Abstracts and Posters

- **T. Zhu**, K. Li, P. Herrero and P. Georgiou, "Recurrent generative adversarial networks for glucose time series generation," in *Diabetes Technology & Therapeutics*, vol. 24, A229-A229, 2022.

- **T. Zhu**, K. Li, P. Herrero and P. Georgiou, "An OpenAI gym environment for reinforcement learning on glucose control in type 1 diabetes," in *Diabetes Technology & Therapeutics*, vol. 24, 2022.

- **T. Zhu**, K. Li, P. Herrero and P. Georgiou, "Blood glucose prediction for type 1 diabetes with population data and model-agnostic meta-learning," in *Diabetes Technology & Therapeutics*, vol. 23, A100-A100, 2021.

- **T. Zhu**, K. Li, P. Herrero and P. Georgiou, "Personalized blood glucose prediction for type 1 diabetes with deep neural networks and attention mechanism," in *Diabetes Technology & Therapeutics*, vol. 23, A100-A101, 2021.

- C. Uduku, **T. Zhu**, K. Li, J. Daniels, P. Herrero, N. Oliver, P. Georgiou, "Independent predictors of hypoglycaemia and impending hypoglycaemia using a wearable physiological data acquisition sensor," in *Diabetes Technology & Therapeutics*, vol. 23, A56-A57, 2021.

- **T. Zhu**, K. Li, C. Uduku, P. Herrero, N. Oliver, P. Georgiou, "Personalized meal insulin bolus for type 1 diabetes using deep reinforcement learning," in *Diabetes Technology & Therapeutics*, vol. 22, pp. A115-A116, 2020.

- J. Daniels, **T. Zhu**, K. Li, C. Uduku, P. Herrero, N. Oliver, P. Georgiou, "ARISES: an advanced clinical decision support platform for the management of type 1 diabetes," in *Diabetes Technology & Therapeutics*, 2020.

- R. Spence, K. Li, C. Uduku, **T. Zhu**, L. Redmond, P. Herrero, N. Oliver, P. Georgiou, "A novel hand-held interface supporting the self-management of type 1 diabetes," in *Diabetes Technology & Therapeutics*, 2020.

Total publications: 23
First-author publications: 17

Abbreviations

AI:	Artificial Intelligence
ANN:	Artificial Neural Network
AP:	Artificial Pancreas
ARIMA	Autoregressive Integrated Moving Average
ARMA	Autoregressive Moving Average
BG:	Blood Glucose
Bi-LSTM:	Bidirectional Long Short-Term Memory
BLE:	Bluetooth Low Energy
BR:	Basal Insulin Rate
BVP:	Blood Volume Pulse
CEG:	Clark Error Grid
CGM:	Continuous Glucose Monitoring
CI	Confidence Interval
CNN:	Convolutional Neural Network
COB:	Carbohydrate on Board
CRNN:	Convolutional Recurrent Neural Network
CSII:	Continuous Subcutaneous Insulin Infusion
CV:	Coefficient of Variation
CVGA:	Control Variability Grid Analysis
DDPG	Deep Deterministic Policy Gradient
DMLP	Deep Multilayer Perceptrons
DNN:	Deep Neural Network
DRL:	Deep Reinforcement Learning

DRNN:	Dilated Recurrent Neural Network
E3NN:	Embedded Edge Evidential Neural Network
EDA	Electrodermal Activity
EHR:	Electronic Health Record
FAPD:	False Alarms Per Day
FCNN:	Fast-Adaptive and Confident Neural Network
FPPD:	False-Positives Per Day
FQE:	Fitted Q Evaluation
GAN:	Generative Adversarial Network
gRMSE	Glucose-Specific Root Mean Square Error
GRU:	Gated Recurrent Units
GUI:	Graphical User Interface
IBI:	Inter-Beat Interval
IOB:	Insulin on Board
IoMT:	Internet of Medical Things
IQR:	Interquartile Range
KNN:	K-Nearest Neighbours
LGS:	Low-Glucose Insulin Suspension
LSTM:	Long Short-Term Memory
MAML:	Model-Agnostic Meta-Learning
MAE:	Mean Absolute Error
MAPE	Mean Absolute Percent Error
MCC:	Matthews Correlation Coefficient
MD:	Mean Deviation
MDI:	Multiple Daily Injection
NLP:	Natural Language Processing
OPE:	Off-Policy Evaluation
OR:	Odds Ratio

PCA:	Principal Component Analysis
PH:	Prediction Horizon
PKM:	Physiologically-Based Kinetic Model
PLGM:	Predictive Low-Glucose Management
PTD:	Prediction Time Delay
RCGAN:	Recurrent Conditional Generative Adversarial Network
ReLU:	Rectified Linear Unit
RFR:	Random Forest Regression
RMSE:	Root Mean Square Error
RNN:	Recurrent Neural Network
SAP:	Sensor-Augmented Pump
SBC:	Standard Bolus Calculator
SMBG:	Self-Monitoring Blood Glucose
SoC:	System on a Chip
SVR:	Support Vector Regression
T1D:	Type 1 Diabetes
T2D:	Type 2 Diabetes
TAR:	Time Above Range
TATR:	Train-on-Augmented and Test-on-Real
TBR:	Time Below Range
TCN:	Temporal Convolutional Network
TD:	Temporal Difference
TD3:	Twin Delayed Deep Deterministic Policy Gradient
TIR:	Time in Range
TL:	Transfer Learning
TRTR:	Train-on-Real and Test-on-Real
t-SNE:	t-Distributed Stochastic Neighbor Embedding
TSTR:	Train-on-Synthetic and Test-on-Real

Chapter 1

Introduction

1.1 Motivation

In the era of artificial intelligence (AI), innovative breakthroughs and emerging technologies are changing the landscape of biomedical and healthcare areas. In recent years, due to the ubiquitous nature of digital information systems and mobile devices in the healthcare industry, a substantial amount of clinical data collected from diverse sources, such as images, text entries, electronic health records (EHRs), sensor measurements, genomics, are available for research purposes. These clinical data offer great promise to develop AI-based applications, especially data-driven approaches, to enhance the treatment of various diseases and accelerate new scientific discoveries. However, the medical datasets from multiple sources are often heterogeneous, high-dimensional, and sparse, and thus they are likely to be underused in clinical scenarios [1]. Fortunately, machine learning, as an increasingly successful AI branch, is powerful at discovering nonlinear correlations of high-dimensional data. The definition of machine learning is that systems are able to learn knowledge and patterns automatically from experience or existing data without being explicitly instructed [2]. Empowered by boosting computational capabilities, a frontier machine learning method, deep learning, has achieved recent success and improved performance surpassing the state of the art in many health domains [3]. Among these, deep learning technology has attracted increasing attention in diabetes research and has been investigated in a rapidly growing number of applications to diabetes management.

Diabetes is a global health problem, defined as a group of lifelong metabolic disorders caused by defective insulin secretion or impaired insulin action. The International Diabetes Federation estimates that there are 463 million people (95% confidence interval (CI): 369–601 million) living with diabetes in 2019, half of whom, however, remain undiagnosed, due to the complex pathogenesis of diabetes [4]. The global prevalence of diabetes is projected to significantly increase in the coming decade. Treating diabetes has been a heavy burden for national economies, healthcare systems, and personal medical expenditures, especially for low- and middle-income countries [5]. The majority of people with diabetes needing exogenous insulin employ the so-called basal-bolus insulin therapy, which consists of measuring blood glucose (BG) levels and delivering insulin with insulin pens or insulin pumps [6].

For people with diabetes, especially type 1 diabetes (TID), it is vital to maintain BG levels in a normal range. Otherwise, hyperglycaemia or hypoglycaemia can cause short and long-term complications in microvasculature and macrovasculature, including neuropathy, nephropathy, retinopathy, stroke, cardiovascular disease, and peripheral vascular disease [7]. However, people with diabetes currently face many real-world challenges and require further improvements in diabetes management systems. First, BG control is a complex intervention, since there are plenty of daily factors influencing BG levels, such as meal ingestion, exercise, alcohol, illness, and stress. Self-management, such as timely BG measurement, insulin, and adherence to recommended lifestyle, is essential, but it requires multidisciplinary knowledge in clinical practice, which is especially challenging for children and adolescents [8]. Due to high inter and intra-population variability in the glucose kinetics process and pharmacokinetics [9], it is difficult to find an optimal therapeutic strategy for all people. An end-to-end digital health system with personalised and intelligent decision-support algorithms is highly desired in diabetes management.

Significant progress has been made in developing continuous glucose monitoring (CGM) for diabetes care. Utilising a minimally-invasive sensor inserted onto the abdomen or the arm, CGM automatically measures subcutaneous glucose concentration and con-

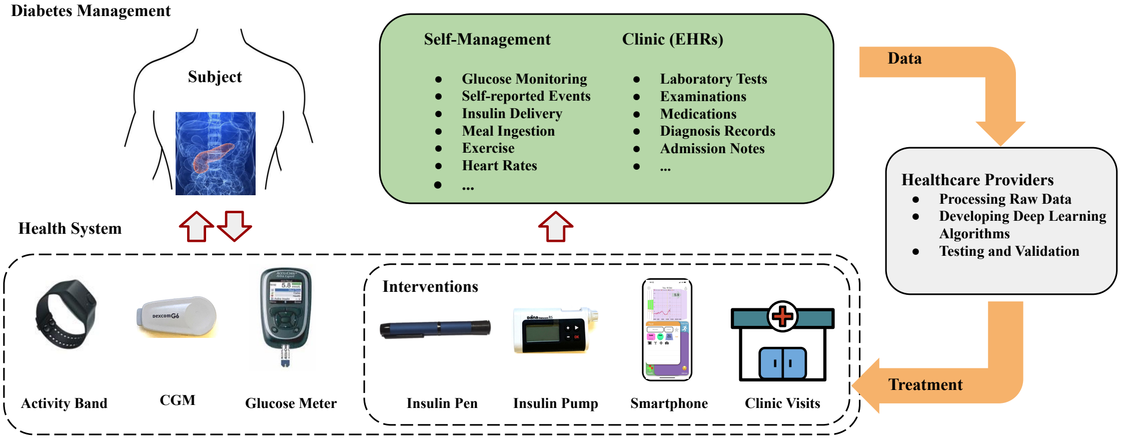


Figure 1.1: Illustration of leveraging data from diabetes management to develop deep learning-based treatment. A large amount of data is generated and processed by healthcare providers to develop deep learning algorithms for the new treatments.

verts the raw signals into BG levels with a fixed frequency (e.g. five minutes). CGM has been demonstrated to reduce the number of severe hypoglycaemic events with multiple daily injections (MDI) [10] and maternal hyperglycaemia [11], which can be combined with an insulin pump as sensor-augmented therapy or hybrid closed-loop glycaemic control [12, 13]. Hybrid closed-loop hormone delivery systems, also known as the artificial pancreas (AP), have been widely researched, aiming at developing automatic glucose regulation. An AP system employs CGM, a control algorithm, and an insulin pump to deliver insulin by continuous subcutaneous insulin infusion (CSII), which has been proven to effectively improve glycaemic control [14]. Smartphone apps to log daily events [15, 16] and calculate bolus insulin are increasingly being adopted and have successfully reduced the daily burden associated with diabetes management. There is also an increasing interest in integrating physical activity monitoring through exercise bands to enhance BG management [17].

The wide use of these wearable devices and digital systems in diabetes management and the increasing EHRs in clinics have produced a considerable amount of available data. This current scenario offers tremendous opportunities to apply advanced data-driven AI methods, especially deep learning, in diabetes research to further improve the treatment and interventions and reduce the risk of diabetes-related complications for people living with diabetes. As depicted in Figure 1.1, a variety of data can be processed by healthcare providers to develop deep learning algorithms for the new treatments. Compared with conventional machine learning technology, deep learning allows the input of raw data and learns representation automatically by exploiting deep neural networks (DNNs), which require little feature engineering work on data pre-processing [18]. Meanwhile, the granular physiological measurements generated in daily management meet the data requirement for deep learning models. Hence, deep learning technology plays a significant role in developing next-generation treatments for diabetes over the last decade [19].

1.2 Research Objectives

The research in this thesis investigates a wide range of cutting-edge deep learning technology and various data features obtained from diabetes cohorts, aiming to make significant steps towards novel personalised decision-support algorithms, wearable devices, and platforms to enhance management and treatment for people living with diabetes. This thesis presents a comprehensive set of deep learning-based methods to address clinical challenges in various scenarios of diabetes management. This thesis also provides complete end-to-end solutions to bring actual therapeutic benefits to people with diabetes, which include data processing, algorithm development, and model implementation in clinical settings. This research is mainly based on BG time series data measured by CGM. Other time series data include the vital signs obtained by wearable sensor wristband (e.g., Empatica E4) and basal insulin levels provided by insulin pumps. In addition, diabetes management apps can provide discrete data of daily events, such as exercise, meal intake and insulin bolus, that can be aligned with time series data as exogenous inputs. Although the research is based on a group of T1D datasets collected in multiple clinical trials, many findings can be generalised to type 2 diabetes (T2D), since more and more people with diabetes, both T1D and T2D, are adopting CGM that enables ambulatory glucose profiles [20].

The main objectives of this thesis and research questions are summarised as follows:

- *Glucose prediction.* Can we develop accurate and reliable glucose prediction algorithms to forecast future BG levels and detect adverse glycaemic events, leveraging DNNs and real-time data? Does the use of non-invasive physiological data improve prediction performance?

- *Glycaemic control.* Can we enhance glycaemic control and maintain BG levels in a target range by controlling hormone delivery with deep learning and reinforcement learning methods?

- *Synthetic data generation.* Is it possible to generate personalised synthetic CGM data to provide ambulatory glucose profiles and increase training data size for better performance of deep learning models in glucose prediction?

- *Digital health systems.* How can we integrate the deep learning algorithms with various digital devices and platforms to be used by individuals, clinicians, and healthcare providers in clinical settings?

1.3 Thesis Structure and Contributions

This thesis is structured into seven chapters. Chapters 3 to 6 present the technical contributions in the order of the aforementioned four research objectives. Figure 1.2 depicts a roadmap of the thesis and relationships between the chapters. An overview of the remaining chapters is presented as follows.

- Chapter 2 introduces the background of diabetes management with various medical devices for glucose monitoring, insulin delivery, and closed-loop control. Then an overview of deep learning technology is presented according to different learning targets and model architectures, which is followed by a systematic literature review of existing deep learning applications in diabetes management. The challenges and current trends in this field are discussed.

- Chapter 3 investigates the prediction of BG levels and adverse glycaemic events. A novel deep learning framework, combing the recent advances of attention mechanism, evidential deep learning, and meta-learning, is proposed. It achieves the state of the

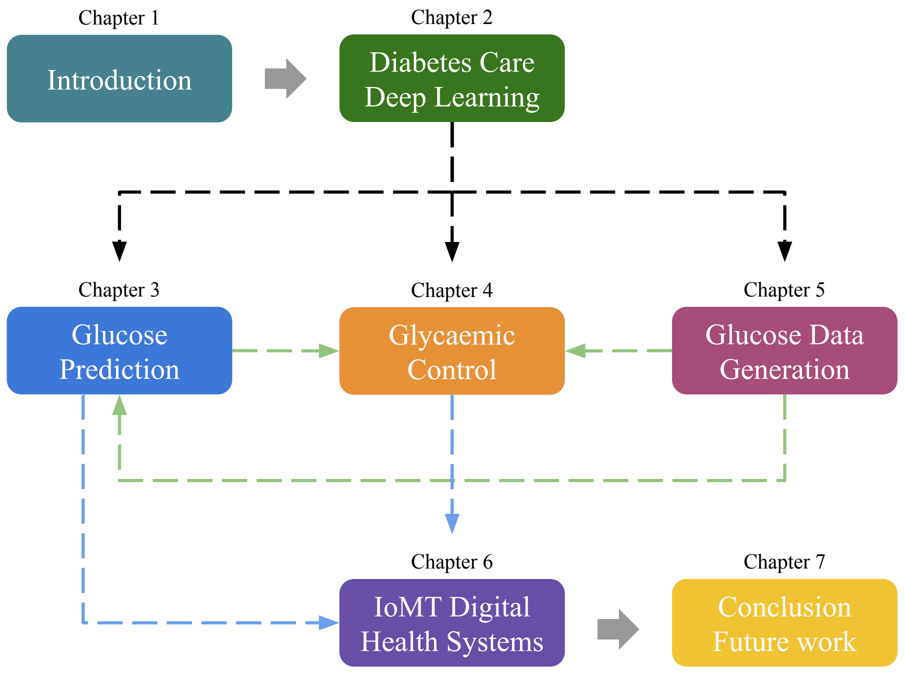


Figure 1.2: Thesis structure and relationships between the chapters. The dashed black arrows indicate the core technology used in each task. The dashed green arrows indicate potential benefits, and the dashed blue arrows indicate the technology deployment.

art in BG prediction and tackles the clinical challenges of model uncertainty and the cold-start issue. The process of model development is presented in detail. The model is evaluated on the three clinical datasets and compared with a group of baseline methods, using a variety of regression, classification, and clinical metrics. The use of non-invasive physiological sensor wristband data collected in a previous clinical trial significantly enhances prediction accuracy.

- Chapter 4 describes the research in glycaemic control using deep learning and reinforcement learning. Problem formulation of glycaemic control in terms of reinforcement learning and the settings of *in silico* simulation are introduced. First, two novel reinforcement learning algorithms are proposed for single- and dual-hormone (basal insulin and glucagon) control and meal insulin bolus recommendation, which significantly improve glycaemic control in an FDA-approved T1D simulator. Next, an offline reinforcement learning framework is introduced to enable control algorithms to be developed and evaluated on actual clinical datasets.

- Chapter 5 investigates synthetic glucose data generation using a novel deep learning-based generative model. Employing the conditional inputs of daily entries of meal, insulin, and self-monitoring blood glucose measurements, the proposed model generates high-quality glucose time series and achieves the best predictive and discriminative scores in the experiments. As a use case, the training data augmented by the synthetic data significantly improve the accuracy of three machine learning glucose predictors for three clinical datasets.

- Chapter 6 presents the integration of deep learning models in actual clinical settings. A customised wristband with a novel low-power and low-cost hardware design is proposed to interact with CGM, forecast BG levels, and send alerts, using an embedded deep learning-based BG predictor that outperforms all the considered baseline methods. Combining edge devices with specifically-designed smartphone, cloud, and desktop platforms, a novel Internet of Medical Things (IoMT) digital health framework is proposed to backup data, visualize historical records, train and fine-tune models, and provide real-time decision support.

- Chapter 7 concludes the thesis by summarising the achievements of this research and discussing the potential directions of future work.

Chapter 2

Background

The aim of this chapter is to provide the reader with the multidisciplinary background of diabetes management and deep learning technology and a systematic literature review of current efforts. To begin with, Section 2.1 to 2.4 introduce diabetes mellitus, glucose monitoring systems, insulin delivery modes, and existing digital health systems for diabetes care and corresponding medical devices that generate data for the development of data-driven algorithms. This is followed by an overview of deep learning technology in Section 2.5, which includes supervised learning, reinforcement learning, and unsupervised learning as the basis of the technical content in this thesis. The results of the systematic review on deep learning for diabetes management are presented in Section 2.6 with the identified challenges and trends described in Section 2.8. The work in this chapter and Chapter 1 has led to the following journal article:

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

2.1 Diabetes Mellitus

According to the etiopathology of diabetes, there are three main clinical categories: T1D, T2D, and gestational diabetes mellitus [21]. Other categories due to specific causes include latent autoimmune diabetes of adulthood and maturity-onset diabetes of the young. Due to the increasing heterogeneity and lack of continuous monitoring, the early diagnosis and classification of diabetes are often difficult in practice [22]. T2D accounts for about 90% of people with diabetes, resulting from insulin resistance or insufficient insulin production. Gestational diabetes mellitus appears during pregnancy and might require lifestyle interventions and exogenous insulin delivery to prevent complications in the infant. T1D is the most common form of childhood diabetes [23], which occurs when the insulin-secreting β-cells of the pancreas are destroyed by the immune system [24].

People with T1D suffer from the absolute insufficiency of endocrine insulin produced by the pancreatic β-cell, hence they rely on exogenous insulin delivery through lifelong management. The primary objective of T1D management is to prevent immediate adverse glycaemic events, including hypoglycaemia and hyperglycaemia, and minimise the risk of long-term diabetes complications. Severe hypoglycaemia may cause seizures, brain damage, and intellectual impairment [25], while hyperglycaemia is a risk factor for cardiovascular diseases, neuropathy, nephropathy and retinopathy [26]. However, intensive treatment in T1D management can be a challenging task for individuals and families, especially for young people, which is associated with a high economic cost [27]. Fortunately, the recent advances in glucose monitoring, insulin therapy, and digital health systems have significantly relieved this daily burden and enhanced the diabetes-specific quality of life [28]. The research in this thesis also particularly focuses on T1D management, aiming to further improve the health and well-being of people with T1D.

2.2 Glucose Monitoring

In general, there are two common glucose monitoring systems in diabetes management: self-monitoring blood glucose (SMBG) and CGM. SMBG is the most conventional and widely used method, which requires finger-pricking to obtain capillary blood samples and uses a BG meter to analyse and report the results [29]. A standard BG meter is a portable device that detects electrical current signals produced by electrochemical test strips, which are proportional to BG concentration.

People with T1D using SMBG tend to finger-prick three to six times per day, but this is usually not enough to present a comprehensive profile of glucose trajectories and effectively prevent undesired glycaemic events [30, 31]. To address this challenge, CGM technology has been developed over the past two decades [32, 33, 34]. CGM might require periodic calibrations based on SMBG measurements, and the most recent systems come with factory calibration [35]. A standard CGM system consists of a minimally-invasive sensor, a non-implanted transmitter, and an external receiver. Using a subcutaneously inserted sensor under the skin, CGM can measure glucose levels in the interstitial fluid and estimate plasma glucose. The sensor usually has a lifespan of 7 to 14 days and can be replaced by the user after use. The transmitter is worn on the skin and attached to the sensor and sends measurements to the receiver. The receiver can be a customized portable device or a compatible smartphone to record and visualise BG levels and other information, such as glucose trends, as well as an insulin pump for feedback on glycaemic control. Depending on how the transmitter communicates with the receiver, there are two types of CGM: real-time CGM and intermittently scanned CGM (i.e., flash glucose monitoring) [36]. Intermittently scanned CGM is usually for newly diagnosed T2D subjects who do not need intensive treatment. Real-time CGM enables time series glucose data and real-time alerts of adverse glycaemic events by measuring BG levels in a fixed time period, e.g., every five minutes, and providing real-time information through a wireless connection. Real-time CGM systems have been demonstrated to improve HbA1c and reduce the number of hyperglycaemia and severe hypoglycaemic events [10, 37, 38, 39], which is recommended in T1D management [39]. Therefore, the sources of glucose data in this research are based on real-time CGM, which is simply referred to as “CGM” in this thesis.

2.3 Insulin Delivery and Artificial Pancreas

The standard therapy of insulin replacement in T1D is known as the basal-bolus insulin regimen [40]. Bolus insulin is administrated at mealtime to compensate for the post-prandial glucose increase, while basal insulin, referred to as background insulin, aims to regulate BG during fasting. This insulin therapy mimics insulin secretion by the pancreas and can be delivered through either MDI or CSII. People on the MDI regimen are given one or two daily injections of long-acting insulin as the basal insulin and mealtime fast-acting insulin as the bolus insulin, and many of them use insulin pens. The amount of insulin is calculated based on BG measurements and carbohydrate content of meal ingestion. Alternatively, CSII involves a wearable insulin pump that delivers fast-acting insulin to subcutaneous tissue. The insulin pump, which can be either a tethered pump or a patch pump, continuously provides a small amount of insulin as basal insulin and delivers meal bolus insulin with a built-in bolus calculator. Compared with conventional MDIs, CSII has been demonstrated to effectively reduce HbA1c, glycaemic variability, cardiovascular mortality, and the severity and frequency of hypoglycaemia frequency in randomised controlled clinical trials [41, 42, 43].

Recent improvements in accuracy and reliability of CGM have accelerated the development of open-loop systems, also known as sensor-augmented pump (SAP) therapy, and hybrid closed-loop systems, also known as AP. Compared with SAP, AP has been shown to decreased hypoglycaemic episodes for various T1D cohorts [44, 45, 46]. An AP consists of, at least, a CGM sensor, a control algorithm, and an insulin pump. Additionally, a dual-hormone AP system also incorporates a glucagon pump to counter-regulate the action of insulin [47, 48]. An AP system is described as "hybrid" because the bolus calculator require manual input of meal content.

Although AP is currently the state of the art in insulin delivery, standard basal-bolus insulin therapy with a capillary BG meter and MDI through an insulin pen remains a cost-effective treatment option. In particular, thanks to the wireless connectivity of these devices to a smartphone, smart pens and smart meters have significantly enhanced this therapeutic option [49]. The datasets used in this research contain both MDI data and CSII data.

2.4 Digital Health Systems in Diabetes Management

In the recent decade, due to the widespread use of smartphones and advances in mobile health, there is an increasing number of diabetes-related smartphone apps available in both iOS and Android markets [50]. However, only a small number of these apps have been evaluated in clinical trials. A few of them have basic decision-support algorithms to calculate bolus insulin and relieve the burden of intensive treatment. The majority of the apps are served as basic electronic diaries and data visualisation tools and lack connections with wearable devices or other diabetes management platforms. Several apps have been reported to enhance the accuracy of carbohydrate estimation and improve postprandial BG control, such as GoCARB with image processing technology [51]. A recent review [52] suggests that, although these smartphone apps could benefit diabetes management in many ways, the design and functionality should better meet the needs of T1D users to improve both clinical outcomes and the diabetes-specific quality of life.

IoMT has opened a door to efficient and reliable BG monitoring and glycaemic control to improve diabetes management [53, 54], leveraging wearable devices and interconnections in AP systems, such as CGM, insulin pumps, insulin pens, glucagon pumps, and physiological wristbands for measuring vital signs (Figure 1.1). In [55], the authors presented a smart diabetes care system with the hardware implementation of a development board and a microcontroller unit to control an insulin pump and transfer health records to cloud storage. They applied a hash algorithm to provide authenticity for individual data and improve the security of the IoMT system. Moreover, Herrero *et al.* [56] proposed the Bio-inspired Artificial Pancreas which comprises a customised handheld unit to implement glycaemic control algorithms on a microcontroller unit and communicate with CGM, insulin pump, and a dedicated smartphone app via Bluetooth. The cloud services were provided in the app for remote monitoring. This system was demonstrated by the UVA/Padova T1D simulator and further validated in a clinical trial [57]. Similar IoMT-enabled AP systems with cloud services have been reported in the literature, such as the Bionic pancreas [47] and DiAs system [58]. Considering various security issues existing in implantable sensors and wireless interconnections, Astillo *et al.* [59] developed a misbehaviour detection system to assess the trustworthiness of the wearable devices in AP systems, including CGM, controllers, and insulin pumps, and also evaluated the system in the UVA/Padova T1D simulator.

2.5 Fundamentals of Deep Learning Technology

Among the wide range of techniques and approaches in deep learning, the overview of popular deep learning methods that are commonly applied to healthcare and, in particular, in the diabetes field is presented. Deep learning originated from artificial neural networks (ANNs) inspired by the structure of biological neurons in the brain [60]. A standard ANN, as depicted in Figure 2.1, comprises a number of nodes and three layers: input, hidden and output layer, to simulate the neuron behaviours by mathematical expressions. In general, an ANN gains the perceptions through an iterative training process called back-propagation but lacks generalization for supervised tasks [61]. By adding more hidden layers, deep learning extends the ANN structure to DNNs for better generalization, which extracts data features and learns representations with thousands or even millions of parameters [62]. The breakthroughs of computational hardware and software infrastructures largely accelerate the development of deep learning by increasing the size and depth of DNN models in the recent two decades [60]. Figure 2.1 depicts five popular DNN architectures employed in diabetes research with the corresponding nodes, cells, and connections. Popular software frameworks to implement deep learning algorithms include Theano [63], Caffe [64], TensorFlow [65], CNTK [66], and PyTorch [67]. These frameworks support various programming languages and hardware acceleration, which help people efficiently build DNN models. In general, most deep learning algorithms can be divided into supervised learning, unsupervised learning, and reinforcement learning.

2.5.1 Supervised Learning

Classification and regression are common tasks in supervised learning, for which the labelled input data is used during iterative model optimization and backward propagation [61]. There are three supervised learning-based DNNs found in the literature on

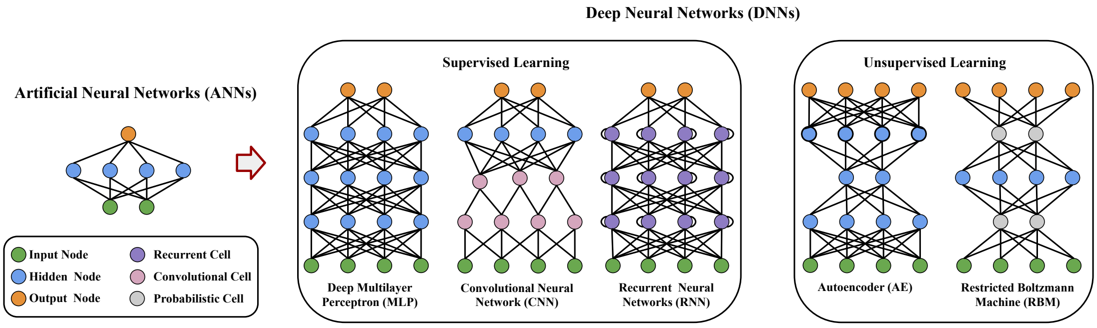

Deep Neural Networks (DNNs)
Artificial Neural Networks (ANNs)
Input Node
Hidden Node
Output Node
Recurrent Cell
Convolutional Cell
Probabilistic Cell
Supervised Learning
Deep Multilayer Perceptron (MLP)
Convolutional Neural Network (CNN)
Recurrent Neural Networks (RNN)
Unsupervised Learning
Autoencoder (AE)
Restricted Boltzmann Machine (RBM)

Figure 2.1: Visualization of ANNs and DNNs. DNNs have an increasing number of hidden layers embedding variants of neural nodes and cells. Higher-level feature maps are computed by deep models. There are five popular DNN architectures in diabetes research: DMLP, CNN, RNN, AE, and RBM.

diabetes: deep multilayer perceptrons (DMLPs), convolutional neural networks (CNNs), and recurrent neural networks (RNNs). The DMLP, also known as a feed-forward neural network, uses the simple connections between neurons, i.e. fully connected layers, and forms the basis of many DNN models. The term “deep” is highlighted to indicate the modes have deep architectures with more than three layers since multilayer perceptrons refer to both ANNs and DNNs in some studies. A DMLP is associated with a set of weight vectors, bias scalars, and nonlinear activation functions, including sigmoid, tanh, and rectified linear units (ReLU) [60].

Leveraging convolutional layers as preceptors, CNNs can process the signals of multi-dimensional arrays and achieve superior performance on imaging tasks [68]. A subsampling layer, or pooling layer, is employed in most CNN architectures to aggregate feature maps. One major advantage of convolutional operations is to reduce the neuron connections between layers, as depicted in Figure 2.1, which notably enhances the efficiency of model training through back-propagation. Empowered by the parallelized operations of graphics processing units and tensor processing units [69], various CNN-based models have been applied to large-scale imaging recognition tasks, such as ImageNet database [70], and transformed into industry practices.

Different from other feed-forward neural networks, the input of an RNN contains the information at the previous timesteps. This feature makes RNNs powerful at processing sequential signals to capture temporal features. However, the difficulty of vanilla RNNs lies in the back-propagation training, where the gradient vanishing and exploding problems are likely to occur [71]. Fortunately, the advanced RNN cells, long short-term memory (LSTM) [72], and gated recurrent units (GRUs) [73], have overcome these problems by introducing gate functions and persevering long-term information. These RNN-based models have provided paradigms in numerous prediction and regression tasks, especially in natural language processing (NLP) and speech recognition. The latest trend in RNN is the attention mechanism [74], which allows models to focus on certain parts of input sequences and map the dependencies regardless of the distances. Chapter 3 focuses on supervised learning techniques, especially RNNs for BG prediction.

2.5.2 Reinforcement Learning

Deep reinforcement learning (DRL) surpasses human professionals in a variety of control problems with high-dimensional environments, where DNNs are employed as the approximators of policy function, value function, or system models. DRL has achieved expert-level control in a number of complex tasks with a high-dimensional environment, such as robotics [75], the game of Go [76], drug design [77], and even nuclear fusion [78]. In general, according to the ways to find optimal control behaviours, DRL can be divided into model-free and model-based approaches. Model-based DRL understands the environment and plans future actions by learning a model from interaction transitions or accessing transition probability distribution. Model-free DRL approaches learn goal-directed behaviours through a trial and error process without the need to model transition probability distribution [79], which optimise control behaviours with a wide range of techniques, such as Q-learning [80] and advantage function. Both model-based and model-free approaches can be developed by policy-based, value-based (i.e., implicit policy), or actor-critic methods, depending on whether the representation of value or policy is built or not.

On-policy learning means the transitions used in the learning process are based on current policy, while off-policy learning can be performed with historical transitions based on previous policy, such as replay memory. Online learning is based on a dynamic dataset, where the old transitions will be immediately discarded or replaced by new transitions in general settings. Offline learning means the algorithms optimise policy with a fixed historical dataset. In most clinical tasks, offline and off-policy DRL is preferred since it does not rely on real-time interactions or explorations, which is safe and inexpensive. Fortunately, the advances of *in silico* trials enable a simulated environment of people with T1D diabetes to develop DRL algorithms. For instance, the UVA/Padova T1D simulator, developed by the University of Virginia (US) and the University of Padova (Italy), is a glucose-insulin dynamics simulator that has been accepted by the Food and Drug Administration (FDA) for pre-clinical studies [81]. Chapter 4 is based on DRL techniques for glycaemic control.

2.5.3 Unsupervised Learning

In unsupervised learning, predefined labels or classes of inputs are not required for the model training. In this context, the algorithm aims at inferring the hidden structures and representations from input datasets without supervision. Unsupervised learning is a powerful tool for data pre-processing, cluster analysis, density estimation, and dimension reduction. The autoencoder (AE) and the restricted Boltzmann machine (RBM) are the two basic architectures. The key feature of AEs is that their training targets are the same as inputs. Latent representations of the input are first transformed by an encoder and then fed to a decoder for reconstruction at the output. An RBM is another approach to mapping the representations by estimating probabilistic distribution over the input data, and thus it is also regarded as a generative model. Compared with standard Boltzmann machines, RBMs only allow the neuron connections that form a bipartite graph, to accelerate training processes. By stacking multiple RBMs, deep belief networks (DBNs) or deep Boltzmann machines can be constructed [82]. In most cases, DBNs are used as feature detectors to extract representations from data by unsupervised learning. However, supervised learning can be further performed to fine-tune the network weights and improve performance for certain learning tasks [83].

A recent breakthrough in unsupervised learning is the generative adversarial network (GAN). A GAN consists of at least two DNNs acting as the generator and discriminator, which are trained in an adversarial process [84]. GANs have been widely used to generate synthetic image data to improve deep learning models for computer vision tasks [85] and have attracted recent attention in time series data generation. Chapter 5 presents a GAN-based framework to generate synthetic BG data.

2.6 Systematic Review on Deep Learning in Diabetes Management

Aiming at identifying and analyzing the benefits of deep learning within diabetes research, we conducted a systematic review by searching multiple public online databases, including PubMed, DBLP Computer Science Bibliography, and IEEEXplore. PubMed is a reputable database for biomedical and clinical research, while DBLP contains millions of publications in the field of computer science. IEEE Xplore is a digital library that covers studies in engineering and allied fields. All three databases provide free and open-access search engines or interfaces without requiring institutional subscriptions like other databases (e.g. Ovid, Scopus, and Web of Science). Therefore, to facilitate the reproducibility of the search results, we have chosen open-access search engines. The initial search based on titles, abstracts, and metadata was performed between April 5 and May 1, 2020, at the beginning of the research in this thesis. We restricted the search to English-language documents that were published between January 1, 2016 and March 31, 2020 (first quarter, Q1), aiming to identify research gaps, and followed preferred reporting items for systematic review and meta-analyses approach [86], as shown in Figure 2.2. Table 2.1 and 2.2 show the selected articles.

Aiming to provide an up-to-date literature review, we further performed a literature search to include the articles published between March 31, 2020, and September 1, 2022, and present them in Table 2.3 and 2.4. The discussion of the latest literature and related work are provided in the introduction sections of the remaining chapters.

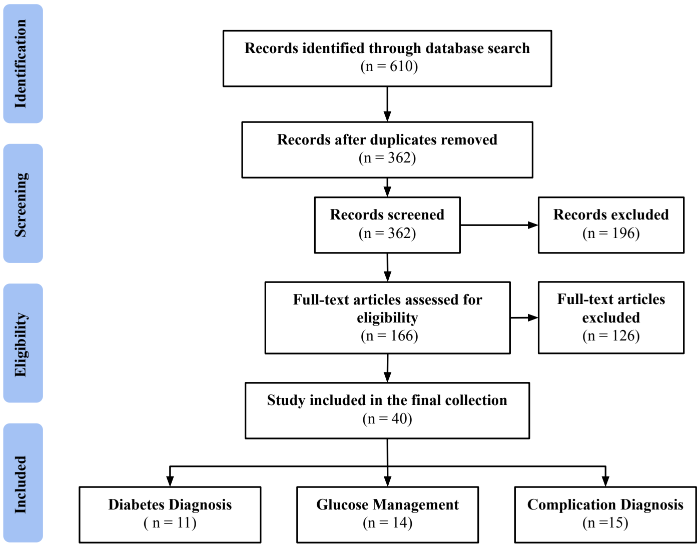

Identification
Screening
Eligibility
Included
Records identified through database search (n = 610)
Records after duplicates removed (n = 362)
Records screened (n = 362)
Records excluded (n = 196)
Full-text articles assessed for eligibility (n = 166)
Full-text articles excluded (n = 126)
Study included in the final collection (n = 40)
Diabetes Diagnosis (n = 11)
Glucose Management (n = 14)
Complication Diagnosis (n =15)

Figure 2.2: Selection process of the preferred reporting items for systematic review and meta-analyses.

2.6.1 Search Strategies

In the literature search, the keywords “diabetes”, “glucose” and “artificial pancreas” were combined with the deep learning terms using Boolean operators AND/OR. The specific query searched was: ((diabetes OR glucose OR artificial pancreas) AND (deep learning OR deep neural network OR convolution neural network OR convolutional neural network OR recurrent neural network OR LSTM OR autoencoder OR Boltzmann machine OR deep belief network)). After obtaining the results of an initial collection of relevant articles, we first excluded duplicated articles from different sources and then performed a manual inspection to evaluate the remaining based on inclusion criteria.

2.6.2 Inclusion and Exclusion Criteria

The studies included in this review are original and available full-text, focusing on deep learning applications in diabetes. The final collection of articles was organised into three categories based on the clinical application: diagnosis of diabetes, glucose management, and diagnosis of complications. Particularly, the included studies were expected to:

- present the details of datasets and data processing
- explicitly describe methods, e.g., the structure of DNNs
- evaluate model performance with standard metrics.

It should be noted that the application of diabetic retinopathy accounts for a large portion of the literature. Thus, in this area, we selected the works presenting DNN results of high originality, or large-scale clinical datasets. Abstracts, posters, technique reports, and reviews were excluded.

2.6.3 Information Extraction

From the selected collection of articles, we inspected the full-text and extracted key information to assess the deep learning applications. The following pre-defined categories were used to present the selected studies.

- **Cases**: We first summarised the specific application cases, i.e. scenarios, of the selected studies to identify the target of each work. For the studies with the available information on the types of diabetes, we have indicated them with † and ‡ for T1D and T2D, respectively.
- **Models**: We present an overview of model architectures which includes a variety of DNN layers and the popular configurations, as mentioned in Section 2.5. The details of hybrid structures and ensemble techniques are also included.
- **Data Sources**: The source of input data is an essential factor for deep learning models. Many studies use more than one dataset, including public and private datasets, to validate the generalization of DNN models. Thus, this category summarises the information regarding the employed datasets, e.g. sources, types, and formats. To facilitate future research to address the issues of data availability, we have highlighted the publicly available datasets with *.
- **Development Process**: This category summarises the strategies for developing

deep learning models, including pre-processing, training, validation, and testing. Although deep learning is good at extracting representations from raw data, these development steps need to be carefully designed, which impact on the functionality and reproducibility of the models.

- **Main Outcomes**: The major outcomes with the corresponding metrics and criteria for performance evaluation, are included in this category. Some of the employed metrics in diabetes and complication diagnosis are sensitivity, specificity, and area under the curve (AUC); and root mean square error (RMSE) and time in range (TIR) is common in glucose management. The results are consistent with the goals in the Cases category.

- **Baselines**: In most selected studies, the authors implemented various baseline methods to compare with the performance of DNN algorithms. Many conventional statistic and machine learning methods are collected in this category, including logistic regression (LR), autoregression (AR), autoregressive integrated moving average (ARIMA), supporting vector machines (SVMs), supporting vector regression (SVR), random forests (RFs), naive Bayes (NB), k-nearest neighbours (KNN), latent variable-based statistical model (LVX), principal component analysis (PCA), and decision trees (DTs). The best performance achieved by the baselines is also presented for the purpose of comparison using the metrics that are consistent with the Main Outcomes category.

- **Limitations**: As a review for an emerging methodology such as deep learning, this category collects the limitations that were identified for the selected studies, which could inspire future work.

2.7 Literature Summary

The initial search yielded a total of 610 papers (PubMed (307), DBLP (31), and IEEE Xplore (272)), as shown in Figure 2.2. After removing the duplicates, 362 papers remained. Then the papers were screened by the inclusion and exclusion criteria. We manually

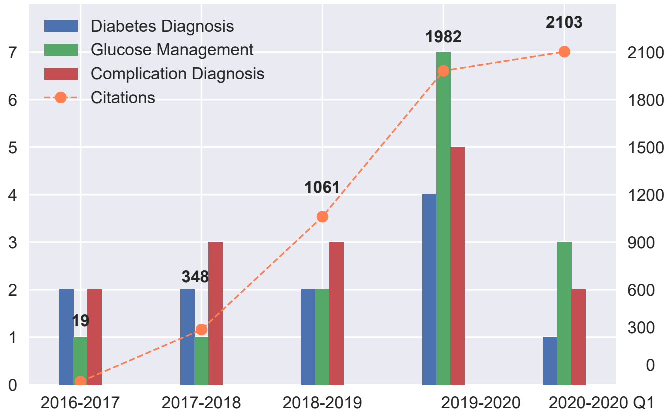


Figure 2.3: Number of articles included in the collection is grouped by the year of publication and application field. The orange dashed line indicates the number of citations in each year.

assessed the eligibility of the remaining papers by full-text inspection and included 40 papers in the final collection. Based on the application scenarios, we divided the final collection into three categories: diagnosis of diabetes (n = 11), glucose management (i.e., diabetes management) (n = 14), and diagnosis of complications (n = 15). As shown in Figure 2.3, most of the selected papers were published in recent years, which indicates that deep learning research for diabetes is a fairly new topic, and its interest has been accelerating. In addition, we also calculated and plotted the number of citations of the selected papers in October 2020, according to Google Scholar. The literature review on glucose management is presented in this chapter (Table 2.1 and 2.2), which is the most relevant to this thesis, while the reviews on the diagnosis of diabetes and diabetes-related complications are presented in Appendix A (Table A.1 and A.2).

The goal of diabetes management is to keep BG levels in the euglycaemia region and avoid undesired glycaemic events (i.e. hypoglycaemia and hyperglycaemia), which can be differentiated: BG prediction, insulin delivery control, and daily-life decision support. Among these, BG level prediction has attracted increasing attention in recent years. An accurate BG prediction enables early interventions to prevent BG anomalies (i.e. hypoglycaemia and hyperglycaemia) and assists SAP therapy (e.g. predictive low-glucose management (PLGM)) and AP systems (e.g. model predictive control) to deliver optimal insulin and/or glucagon doses. The use of smartphone apps allows people to report the

Table 2.1: Literature review on deep learning in diabetes management (Part I)

Ref.	Cases	Models	Data Sources	Development Process	Main Outcomes	Baselines
[87]	Detect hypo†	DBN by stacking RBMs	15 T1D children monitored for 10-hour overnight	Calculating QT correlation and heart rate; training and testing data: 10 and 5 subjects	Sensitivity: 79.70%, specificity: 50.00%	DBN by ANNs (76.28%, 50.40%), multiple regression
[88]	Predict BG levels†	LSTM and a linear layer	A clinical database with the T1D data over 1600 days	Pre-training; linear interpolation; training and testing data: 5 patients in 400 days, another 5 patients with 200 samples	RMSE for 30, 60-min PH: 21.4, 38.0 mg/dL	ARIMA, EPM with SVR (21.6, 39.2 mg/dL)
[89]	Predict BG levels†	LSTM, Bidirectional LSTM and 3 FC layers	(1) GoCarb dataset (20 adults) [90], (2) UVA/Padova T1D simulator (11 adults) [81]	Pre-training on (2); linear interpolation on (1); cross-validation (67%, 33%); testing data: 26 sub-dataset from (1), 1791±141 CGM samples	RMSE for 30, 45, 60-min PH: 11.63, 21.75, 36.92 mg/dL	ARIMA, SVR (11.69, 22.14, 37.42 mg/dL)
[91]	Predict BG levels†	Deep sequential polynomial model (RNN)	40 T1D subjects over 1900 days	The ratio of training,validation and testing data: 85%, 7.5% and 7.5% from 555,000 CGM samples	Absolute percentage error for 30-min PH: 4.87	Linear extrapolation, RF, RNN(LSTM, 5.3)
[92]	Glycaemic control†	CNN (Inception-v3)	Food-101 dataset (101 classes, 101,000 images) [93]⋆	Image augmentation; pre-training on ImageNet dataset; training and testing data: 75,750, 25,250 images	TIR: 91.76%, top-1 accuracy of the image classification: 81.65%	Standard controller in UVA/Padova T1D simulator (TIR: 78.8%)
[94]	Predict BG levels‡	LSTM with dynamic time warping	A dataset from a randomized trial (26 adults, smartphone group (n = 11))] [95]	Pre-processing; TL; min-max normalization across patients; training, validation and testing data: CGM samples of 120, 30 and 30days	Clark Error Grid zones of next-day PH (A: 84.12, B: 15.16, C: 0, D: 0.72, E: 0)%	ANN, KNN, ridge regression (A: 83.03%), kernel ridge regression, moving average
[96]	Predict BG levels†	CNN, LSTM and 2 FC layers	(1)UVA/Padova T1D simulator (10 adults) [81], (2) 10 clinical subjects	Using Gaussian filter to remove outliers; training and testing data: 50% and 50%	RMSE for 30, 60-min PH: 9.38, 18.87 (1); 21.07, 33.27 (2) mg/dL	SVR (22.00, 34.35 (2)), LVX, neural network, AR
[97]	Predict BG levels†	LSTM and a FC layer	RT_CGM dataset (the population of 451 patients) [98]⋆	Removing sequences with low quality; Tikhonov regularization; training and testing data: 304,450 and 94128 samples	RMSE for 30, 45, 60-min PH: 19.47, 26.47, 32.38 mg/dL	AR, ANN, standard RNN, non-linear AR (24.66, 32.33, 38.58 mg/dL)
[99]	Predict BG levels†	Memory-Augmented LSTM with neural attention weights	(1) OhioT1DM dataset [100]⋆, (2)AIDA Simulator, (3)UVA/Padova T1D simulator [81]	Pre-training; linear interpolation and extrapolation; testing, validation and training data: last 10 days, previous 10 days, rest days (1); 400, 100, and 100 days (2); 70, 10 and 10 days (3)	RMSE for 30, 60-min PH: 18.74, 30.63 (1); 1.23, 2.27 (2); 2.93, 4.92 (3) with input of CGM, insulin and meal events	ARIMA (20.17, 33.47 (1); 5.59, 16.48 (2); 12.00, 18.66 (3))

† T1D, ‡ T2D, ⋆ publicly available dataset

Table 2.2: Literature review on deep learning in diabetes management (Part I cont.)

Ref.	Cases	Models	Data Sources	Development Process	Main Outcomes	Baselines
[101]	Predict BG levels†	Dilated CNN (residual and parameterized skip connections)	(1) UVA/Padova T1D simulator [81], (2) ABC4D dataset, (3) OhioT1DM dataset [100]*	Ruling out outliers; interpolation/extrapolation; label transformation; training, validation and testing set: 45%, 5%, 50% (1, 2); 40 and days (3)	RMSE for 30, 60-min PH: 8.88, 19.90 (1); 19.19, 31.78 (2); 19.28, 31.83 (3)	SVR (21.75, 34.31 (3)), LVX (12.25, 22.41 (1)), neural network (20.42, 33.13 (2)), AR
[102]	Glycaemic control†	Deep Q-network with GRU or 1-D CNN	UVA/Padova simulator (30 virtual subjects) [103]	Using CGM and insulin data in past 24 hours as the states, action setting: {0, basal rate, 5*basal rate}; testing in 10 days	Average risk index for the virtual subject: 9.26	Proportional-integral-derivative control (11.80)
[104]	Predict BG levels†	2 branches of LSTM cells (past and future information)	(1) UVA/Padova T1D simulator (100 adults) [81], (2) Padova clinical dataset (1 patient) [105]	Min-max normalization; output filtering in (2); training data: four-day protocol (1), testing data: 3-day scenario and *in vivo* data over a month (2)	Average RMSE for PH of 60 minutes: 11.72 (1), 21.09 (2)	Linearized average model (46.82 (2)), daily model predictor
[106]	Predict HbA1c†	1-D CNN, Inception module, FC layers	A clinical dataset (759 T1D subjects, 1543 observations)	Behavioral feature extraction; manual feature extraction; 10-fold cross-validation; batch normalization; loss regularization	Mean absolute error: 4.80, the coefficient of determination: 0.71	Nathan's formula, CNN (5.98, 0.62), manual features extraction network
[107]	Predict BG levels†	LSTM and 2 FC layers	OhioT1DM dataset [100]*	Scaling glucose values by 0.01; the ratio of training, validation and testing data: 60%, 20% and 20%	RMSE for 30, 60-min PH: 18.867, 31.403	Previous work with machine learning

† T1D, ‡ T2D, * publicly available dataset

Table 2.3: Literature review on deep learning in diabetes management (Part II)

Ref.	Cases	Models	Data Sources	Development Process	Main Outcomes	Baselines
[108]	Predict BG levels†	LSTM and a FC layer	(1) OpenAPS data (55 participants) [109], (2) OhioT1DM dataset [100]⋆	Removing outliers; linear interpolation and median filtering (training); linear extrapolation (testing)	RMSE for 30-min PH: ; 14.53 (1); 19.12 (2) mg/dL	LR (15.33(1), 19.59 (2)), SVR, RF, ARIMA
[110]	Glycaemic control‡	CNN and FC layers	In silico data simulated by Medtronic virtual patient model [111]	Engineered features and plasma glucose measurements; 43740 adherent days and 43740 nonadherent dayas	Accuracy: 79.8%	LR (78.6%), ANN
[112]	Predict BG levels†‡	CNN with adversarial TL	(1) IDIAB dataset (T2D), (2) OhioT1DM dataset [100]⋆, (3) T1DMS dataset [81]	Standardization; linear interpolation and extrapolation; training, validation data (80%, 20%); testing data (5 days (1), 10 days (3))	RMSE for 30-min PH: 19.61 (1); 19.45 (2) mg/dL	SVR (20.32 (1), 20.10 (2) mg/dL)
[113]	Glycaemic control†	N-BEATS	OhioT1DM dataset [100]⋆	Meal correction; linear interpolation; sample filtering	RMSE for cards recommendation: 9.79; bolus: 0.85	LSTM (11.15, 1.02)
[114]	Predict BG levels†‡	GRU, CNN, and Transformer with TL	(1) Beth Israel Deaconess Medical Center (40 T2D outpatients) (2) OhioT1DM dataset [100]⋆	Pre-training; data augmentation; leave-one-out cross-validation	RMSE for 30 and 60-min PH: 19.08, 33.80 (2) mg/dL	LSTM with attention (18.82, 31.10 (2) mg/dL)
[115]	Glycaemic control‡	Double deep Q-network	Singapore Health Services Diabetes Registry (189,520 T1D subjects) [116]	Training, testing data (80%, 20%); 49-dimension state vector; prioritized experience replay;	Model-concordant treatments: better control (OR 1.73, 95% CI 1.69-1.76)%	Model-nonconcordant treatments
[117]	Predict BG levels†	CNN and LSTM with multitask learning	OhioT1DM dataset [100]⋆	Two-hour sliding window; Linear interpolation and extrapolation	RMSE for 30, 60-min PH: 18.8, 31.8 mg/dL	SVR (19.2, 32.6)
[118]	Predict BG levels†‡	CNN	Samsung Medical Center (1,114 subjects) [119]	Min-max normalization; training, validation, and testing data (64%, 16%, 20%)	RMSE for 30, 60-min PH: 17.8, 28.1 (T1D); 17.2, 27.7 (T2D) mg/dL	RF (18.6, 29.2 (T1D); 17.7, 28.6 (T2D))
[120]	Predict BG levels†	Dilated CNN	Tidepool Big Data (97 T1D subjects) [121]	Piecewise cubic interpolation; Average lengths of training and testing data: 143, 25 days	RMSE for 30-min PH: 23.2 mg/dL	Kernel ridge regression (25.77), Gaussian process

† T1D, ‡ T2D, ⋆ publicly available dataset

Table 2.4: Literature review on deep learning in diabetes management (Part II cont.)

Ref.	Cases	Models	Data Sources	Development Process	Main Outcomes	Baselines
[122]	Predict BG levels†	Vanilla LSTM with ensembles	OhioT1DM dataset [100]*	Linear extrapolation; training, validation data: 80%, 20%	RMSE for 30, 60-min PH: 19.83, 34.21 mg/dL	LR (19.85, 33.63)
[123]	Synthetic data†	Generative adversarial network	Hospital clinic of Barcelona [124], OhioT1DM dataset [100]	Linear interpolation; data reshaping; data augmentation for hypoglycaemia prediction	Sensitivity: 69%, specificity: 79%	Previous work on machine learning for hypoglycaemia prediction
[125]	Glycaemic control†	Proximal policy optimization with LSTM	UVA/Padova simulator (30 virtual subjects) [103]	Current CGM as state; blood glucose risk index rewards; insulin action between 0 to 30 units	Smallest risk index, euglycemia is maintained in 73%	Basal-bolus and PID control

† T1D, ‡ T2D, * publicly available dataset

exogenous events that influence BG levels. By temporally aligning CGM measurements with these self-reported events, such as meal composition and insulin dosage, a multivariate time series can be formed and processed by deep learning models. Normally, the prediction horizon (PH) for short and long-term forecasting is 30 minutes and greater than 60 minutes respectively. In this scenario, the RNN-based architecture is a powerful tool, referring to its success in temporal sequence processing and regression. Augmented by LSTM cells, the RNN is the most widely used method for glucose prediction in Table 2.1. Mirshekarian *et al.* proposed an LSTM model for 30 and 60-minute prediction, which outperforms the engineered physiological model (EPM) with SVR [88]. EPM is a continuous dynamic model used to calculate the system states, which comprises the compartments of meal absorption dynamics, insulin absorption dynamics, and glucose-insulin dynamics. They further introduced a neural attention layer to emulate the case-based prediction by a memory module [99].

In addition, Li *et al.* transferred the prediction into a classification problem and used 1-D dilated CNNs in their GluNet framework to classify the predictive changes of future BG values [101]. The study tested the model on the two clinical datasets. The use of dilated DNNs to improve the BG prediction is also highlighted in [126, 127, 128]. Similarly, Zaitcev *et al.* employed 1-D CNNs with the Inception module to estimate HbA1c from imperfect time series of CGM [106]. An important consideration for BG prediction is whether the algorithm can be applied in real time. In this regard, the deep learning models based on CNN and LSTM layers have been validated in smartphone apps to perform real-time BG prediction with short inference time and small memory consumption [96, 101].

Another application of CNNs is to estimate macronutrient content as a daily-life support [92]. The trained CNN model can predict the food category based on the food images from smartphones and then assist decision support systems and AP systems to compute the required amount of meal bolus insulin. The proposed algorithm was validated in the UVA/Padova T1D simulator with the disturbances of carbohydrate content and incorrect estimation of meal sizes. In recent years, many research groups use computer simulation, i.e. *in silico* setup, to test algorithms in various virtual scenarios, considering the high costs and safety concerns associated to actual clinical trials in humans and animals. Fox *et al.* used the UVA/Padova T1D simulator to test DRL algorithms to control the delivery of basal insulin, using GRU and 1-D CNN architectures.

Notably, although many of the studies conducted experiments on their proprietary clinical datasets, there are two datasets publicly available to researchers, the OhioT1DM dataset and the RT_CGM dataset. The OhioT1DM dataset was released for the first edition of the BG level prediction challenge in 2018 and later updated for the 2020 edition [100]. It contains multi-modal data with a total of 20 data fields corresponding to 12 subjects with T1D over eight weeks. Among these, BG levels were the most important feature and were recorded by Medtronic Enlite CGM every five minutes. Self-reported events recorded by a smartphone app, including carbohydrate estimate for meal intake, insulin bolus amount, and duration and intensity of exercise, were widely used in the selected articles and can be combined with CGM data for multivariate time series input. Other physiological data were measured by the Basis Peak band or Empatica Embrace band and aggregated every five minutes. In the RT_CGM dataset, only CGM is available, corresponding to 451 subjects with T1D over 26 weeks. There are unavoidable errors in real-world data. The sources of errors in recent CGM systems (e.g., Dexcom G6) include the conversion of interstitial measurements into the vascular space, imperfect calibration, and random measurement noise [129]. The mean absolute percentage deviation between

CGM and a BG meter is typically around 5-10%. Manual data entry errors, such as missing input and misestimation, are inevitable in self-reported events. Meal carbohydrates are often underestimated by around 20% of total carbohydrate content [130]. To this end, the UVA/Padova T1D simulator [81, 103] is frequently employed to generate synthetic population datasets for additional validation, which can provide CGM, meal, and insulin data. The users are allowed to control the aforementioned errors by adjusting the parameters of the models that simulate medical devices and daily scenarios.

Furthermore, an unsupervised learning algorithm based on DBNs, and taking ECG signals as input, was employed to detect hypoglycaemia in children with T1D [87]. Similarly, in a recent study, ECGs were used to detect nocturnal hypoglycaemia in healthy individuals with a CNN-LSTM model [131].

2.8 Challenges and Current Trends

Although deep learning has improved the state of the art in several areas of diabetes, the applications in healthcare systems need to be robust, reliable, and convincing to avoid safety issues and provide effective therapeutic aids. In this context, there remain several limitations and challenges for deep learning to be further introduced in actual clinical settings. Two main limitations identified from the selected articles are summarised as follows.

- **Model issues**: The model uncertainty represents the confidence of model outputs and indicates whether the decisions made by the model are reliable and safe in healthcare settings [132]. Interpretability, i.e. explainability, stands for how the model obtains the corresponding output based on a set of inputs. In many cases, deep learning models are regarded as "black boxes" with a lack of model transparency due to complex nonlinear layersa [96]. As a consequence, if the model performance degrades in certain circumstances, it might be difficult to explain why. It is also challenging to implement increasingly complex models on current medical devices and platforms. Solving these issues is an important goal for AI in healthcare to convince clinicians to adopt such systems.

- **Data issues**: Training a deep model for complicated tasks requires a high volume of data [94, 107]. Collecting data from people with diabetes is often time-consuming and expensive, compared to other tasks in computer vision and NLP. Consequently, many studies face a shortage of data during their research cycles. The variability among people with diabetes is largely due to complex glucose dynamics. To obtain better generalization for deep learning models, the training data needs to cover a diverse range of individuals, such as people of different ages and comorbidities [87, 92, 94, 97, 106]. However, many datasets are often collected from a specific cohort of people, which lacks diversity and could bring bias to learning. Similar to many other problems in healthcare, most diabetes datasets are heterogeneous, sparse, and noisy with some missing values [94, 101, 102, 106, 107]. It is not realistic to collect perfect data from either clinical practice or daily self-management, e.g., the unavoidable errors from CGM sensors.

In real-world scenarios, the data collected from people with diabetes are prone to be imperfect, due to human errors and sensor artifacts. The process to collect real data is sometimes expensive and time-consuming. Due to data privacy policies, sharing datasets among research teams is sometimes difficult. These factors lead to many studies employing a reduced, sometimes insufficient, amount of data. Another challenge that arises due to the complexity of glucose dynamics is how to process the available data in order to characterise people with diabetes. Also, deep learning models lack transparency. From the perspective of clinicians, why the models produce the output for a certain input case and whether the output can be trusted or not are important, particularly for some critical decision-making applications. The complicated structures in DNN layers can effectively learn the patterns from non-linear signals but reduce the interpretability the model. The introduced challenges not only apply to the field of diabetes but also are valid in other health domains. Fortunately, the recent development in both diabetes care and deep learning can provide new solutions to meet these challenges.

First, digital records and vital signs are increasingly collected by multi-modal systems with wearables and smartphone apps. Most of these data are conveniently uploaded to centralised systems or cloud repositories. With the popularisation of IoMT and 5G networks, data volumes and variability of data sources are expected to significantly increase in many healthcare applications, and in particular, in diabetes care. As the data volume expands, many low-quality data samples can be filtered out and removed from training sets, and the advances in wearables (e.g. CGM) can effectively reduce measurement errors. Deep learning is well adapted to cope with such an increase in data availability. Several publicly available datasets are outlined above, and more datasets will be shared in the communities after proper post-processing and anonymisation. In order to deploy deep learning in an ambulatory setting, the software frameworks mentioned in Section 2.5 can be easily ported to mobile devices by using tools such as TensorFlow Lite [96, 101].To interpret deep learning technology in healthcare, many recent attempts in the AI domain have been made to enhance model transparency and understand model functionality. In particular, a unified framework, the Shapely additive explanations, was proposed to explain the input features that contribute to the final output, which has been validated in many data-driven healthcare applications [133]. This is also an effective method to select input features by ranking their importance.

Instead of solely using data-driven models, integrating the expert knowledge in the learning process can help to better understand the underlying mechanisms of a health condition, such as diabetes. Specifically, there are two feasible methods. One is to incorporate the physiological parameters as the input feature of the models, and the other is to use expert knowledge as a guide during the training process. Expert knowledge is also essential to craft safety constrains and calculate the confidence of the model outputs.

Training a very deep model from scratch is time-consuming since millions of parameters in the DNN units need to be tuned. In this regard, an approach called transfer learning (TL), i.e. pre-training, provides a shortcut to solve this issue. In the tasks of BG prediction, the *in silico* datasets derived from the simulators and a portion of real clinical data are used for pre-training [88, 89, 99]. It is an effective method to mitigate the high demand for data during the DNN training. Alternatively, data augmentation is a way to improve model performance with limited data [114].

In this research, we addressed the aforementioned challenges as follows. For the model issues, Chapter 3 provides theoretically-supported upper and lower bounds to indicate model uncertainty and corresponding confidence intervals, while a customised wristband and a smartphone app are proposed in Chapter 6 for model implementation. To train a deep learning model with limited data, we present a meta-learning framework in Chapter 3 to enable few-shot learning in personalised BG prediction, and Chapter 4 uses a TL framework to generalise the glycaemic control algorithms from T1D simulators to clinical settings with a small number of interactions. Chapter 5 further introduces a GAN-based framework to generate synthetic BG time series to increase the amount of personalised data.

2.9 Conclusion

This chapter provided an overview of the clinical settings in diabetes management and the fundamentals of deep learning technology. After performing a rigorous search to select a collection of articles and summarise the key information, we presented a comprehensive systematic review of the current trends in deep learning technology for diabetes management. A significant number of deep learning methods have been adopted by the diabetes research community, covering various DNN architectures that outperformed previous conventional machine learning approaches. Moreover, challenges in medical data and deep learning models have been identified in the literature.

Deep learning is a hotspot in the era of AI, and it is worth noting that most of the selected papers are publications from the recent years (Figure 2.3), which indicates that this is an emerging technology. Hence, there is great potential to meet the aforementioned challenges and improve the current applications by transferring the latest advances in deep learning into massive multi-modal data on diabetes management, as one of the objectives of this thesis. It is observed that most of the existing studies focused on BG prediction. Therefore, this thesis also aims to develop novel deep learning-based methods and address practical considerations in other clinical scenarios of diabetes management. We expect that deep learning technology will bring actual therapeutic benefits and largely improve the treatment for people living with diabetes.

Chapter 3

Deep Learning for Personalised Blood Glucose Prediction

The availability of large amounts of data from CGM, together with the latest advances in deep learning technology, has opened the door to a new paradigm of algorithm design for personalised BG prediction in T1D with superior performance. However, several challenges prevent the widespread implementation of deep learning algorithms in actual clinical settings, including unclear prediction confidence and limited training data for new T1D subjects, as identified in the previous chapter (Section 2.8). To this end, a novel deep learning framework, Fast-adaptive and Confident Neural Network (FCNN), is proposed in this chapter. The proposed framework is first validated on the OhioT1DM, ARISES, and ABC4D datasets with standard input data of CGM and self-reported events. FCNN significantly improved prediction performance for 30 and 60-minute PHs in the experiments and successfully addressed the aforementioned challenges.

The rest of this chapter investigates the efficacy of non-invasive wristband sensor data available in the ARISES dataset collected in a six-week longitudinal clinical study. Several significant associations between wristband sensor data and adverse glycaemic events are identified in statistical analysis. We proceed to develop a prediction model based on the FCNN framework, which utilises multi-modal data from CGM, a smartphone app, and the sensor wristband to predict glucose levels and hypo- and hyperglycaemia. When compared with baseline methods, the proposed algorithm achieved the smallest error for predicting glucose levels and detecting hypoglycaemia and hyperglycaemia. The work in this chapter has led to the following journal articles:

- **T. Zhu**, K. Li, P. Herrero, P. Georgiou, “Personalized blood glucose prediction for type 1 diabetes using evidential deep learning and meta-learning,” *IEEE Transactions on Biomedical Engineering*, 2022.

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

3.1 Introduction

Accurate BG prediction is a valuable tool for enhancing decision support systems in diabetes care, which aims to mitigate these adverse glycaemic events and reduce the burdens on people living with diabetes. In particular, BG prediction enables proactive interventions, such as glucose alerts in CGM and the PLGM with basal insulin suspension in SAP or AP systems [134]. CGM allows for real-time tracking of BG levels and has been verified as an effective device for BG control in T1D management [33, 34]. There is a rising trend of connecting CGM with smartphone apps to display retrospective BG trajectories and allow users to record daily events (e.g. meals, insulin doses, exercise) that have an impact on glucose levels. The wide use of CGM and mobile apps has produced a large amount of data, which enables the development of data-driven BG prediction [135, 136, 137], especially machine learning algorithms. Among these, empowered by DNNs, deep learning algorithms have achieved superior performance [19, 126, 127]. The accuracy of BG prediction is essential to decision making in clinical settings. In this regard, a common method to evaluate clinical accuracy is error grid analysis (Section 3.4), e.g., Clark error grid (CEG) [138]. CEG shows a scatter plot of ground truth versus predictions and divides the grid into five zones. The diagonal stands for perfect predictions, while values in the A and B zones are clinically acceptable and would not lead to inappropriate treatment. For a CGM sensor measuring BG levels every five minutes, a 1% increase in A and B zones can avoid approximately three inappropriate interventions per day. Therefore, more than 95% of predictions within A and B zones are needed. In addition, BG predictions with low RMSE can benefit AP systems using predictive control algorithms that calculate insulin doses based on the predictive value at each timestep [139].

Although deep learning approaches have improved the BG prediction accuracy, there are still many challenges in translating these models into clinical settings (Section 2.8 in Chapter 2). According to feedback from clinicians and clinical outcomes reported in previous studies [3, 19, 140], we identified two clinical challenges that are most significant:

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

Aiming to tackle these challenges, we propose FCNN with attention-based RNNs for personalised BG prediction. Figure 3.1 depicts the real-world settings of a T1D management system. Given the multivariate features accessible from wearable devices and a smartphone app, a DNN model can be developed by the FCNN framework that incorporates the latest advances in deep learning to address the above challenges. To the best of our knowledge, this is the first work that adopts evidential deep learning and meta-learning in BG prediction. The experiments show that FCNN exhibits excellent accuracy and outperforms all the considered baseline methods.

A recent breakthrough in both NLP [141] and computer vision [142] is the attention mechanism. In particular, it allows an RNN model to focus on the hidden states and learn long-term temporal dependencies at certain timesteps by calculating a context vector that is weighted by alignment scores. Various attention mechanisms and corresponding score functions were proposed in the recent literature [143], including scaled dot-product [74], additive [141], general [144], dot-product [144], and location-based form [144]. The trans-

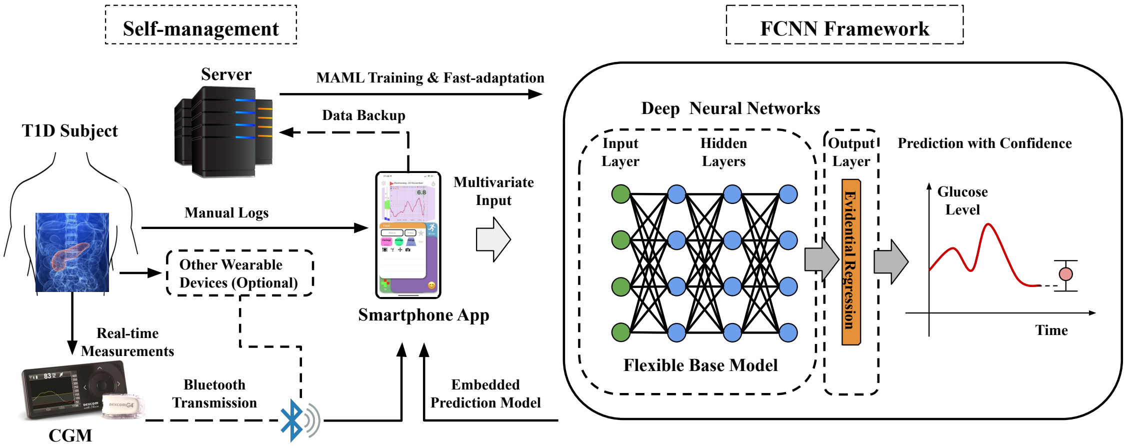


Figure 3.1: System architecture of incorporating FCNN in a T1D management system. A smartphone app receives BG measurements from CGM, data from other wearable devices (e.g., insulin pumps and wrist bands), and records of relevant daily activities. After uploading the multivariate data to a server (e.g., cloud repository), FCNN is used for fast adaptation of a deep learning model with confidence. Then the well-trained model is embedded into the smartphone app to perform real-time prediction. Besides the bidirectional RNN proposed in this chapter, FCNN also supports other customised DNNs as base models.

former is based on a multi-head self-attention mechanism, which is a highly successful deep learning architecture in recent NLP studies (e.g., BERT and GPT-3) [145]. In the context of personalised BG prediction, Mirshekarian *et al.* [99] presented a memory-augmented LSTM using the attention weights derived by a two-layer dense network (i.e., additive form), which exhibited improvement on synthetic data. In this chapter, we adapt another sequence-to-sequence attention mechanism (i.e., general form) for many-to-one prediction, according to model validation performance. In addition, we employ the transformer as a baseline method in the experiments.

Taking advantage of historical datasets, TL approaches have been widely used in previous studies to improve personalised BG prediction performance with pretraining and fine-tuning [99, 128, 146]. To further reduce the demand for individual data in the fine-tuning phase, meta-learning, also referent to as learning to learn, is an emerging approach for optimising a meta-model across a set of learning tasks, so that fast adaptation can be performed with only a few data samples. In particular, model-agnostic meta-learning

(MAML) [147] can learn the initialisation of a DNN model with good average performance and has already been applied to assist clinicians in sleep stage classification [148] but has never been used in the context of BG prediction. Treating each T1D subject as a learning task, We use MAML and compared it with a conventional TL method.

Most of the existing work treated BG prediction as a traditional regression task and used mean square error as the loss function to obtain a single prediction value [19]. In [107], a mixture density network was proposed to model the uncertainty with a univariate Gaussian distribution. However, this configuration can only model data uncertainty by a mixture distribution [149]. Here, we incorporate evidential regression [150] to simultaneously map data uncertainty and model uncertainty.

3.2 Fast-Adaptive and Confident Neural Network

3.2.1 Problem Formulation

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

$$\mathbf{X}_t = [\mathbf{x}_{t-L+1}, \mathbf{x}_{t-L+2}, \ldots, \mathbf{x}_t] \in \mathbb{R}^{d \times L}, \tag{3.1}$$

where $\mathbf{x}_t \in \mathbb{R}^{d \times 1}$ contains d features at the timestep t, and L is the window length, i.e., the number of timesteps of the input. CGM series are obtained from real-time sensor measurements, while the amount of carbohydrates and insulin are based on the daily records from mobile apps.

Given a target PH and CGM resolution τ, the BG level at timestep $t+r$ is expressed as G_{t+r}, where $r = \mathrm{PH}/\tau$. To reduce bias [101, 128, 151], we use the signed difference between the current and future BG level as the prediction targets, i.e., $y_t = f_N(G_{t+r} - G_t)$, where f_N is the min-max normalization function to scale data within a range of $[0, 1]$.

Table 3.1: List of data features

Feature	Description
CGM*	BG time series measured by CGM (mg/dL)
Carbohydrate*	Carbohydrate intake of meal ingestion (g)
Bolus insulin*	Amount of bolus insulin delivery (unit)
Basal insulin	Amount of basal insulin delivery (unit)
COB	Carbohydrate on board derived by Equation (3.3) (g)
IOB	Insulin on board derived by Equation (3.3) (unit)
Time index	Normalised 24-h time with a range of [0, 1]

* Feature selected for the model input

Hence, the BG prediction $\hat{G}_{t+r}$ can be defined as

$$\hat{G}_{t+r} = f_N^{-1}(\hat{y}_t) + G_t \tag{3.2}$$

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

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

Table 3.1 summarises the feature set extracted from the clinical datasets, including CGM measurements, bolus insulin, basal insulin, the ingested carbohydrate amount, insulin on board (IOB), carbohydrate on board (COB), and time index series. Bolus insulin is used to compensate for BG increase after meal ingestion and hyperglycaemia during fasting, while basal insulin aims at maintaining BG levels in a target range when fasting. IOB and COB, which are commonly employed in bolus calculators and AP, were derived from linear models [152] and formulated as follows.

$$\begin{aligned} \mathrm{IOB}_t &= B * \max(0, 1 - \frac{t - t_B}{T_{\mathrm{IOB}}}), \\ \mathrm{COB}_t &= \max(0, C - R_{\mathrm{COB}} * (t - t_C - \Delta_{\mathrm{COB}})), \end{aligned} \tag{3.3}$$

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

3.2.2 Model Architecture

Figure 3.2 shows the overall structure of the proposed model. We instantiate a stack of three bidirectional GRU layers to extract feature maps from the multivariate input. The complete list of notations can be found in Appendix B (Table B.1). The computation of

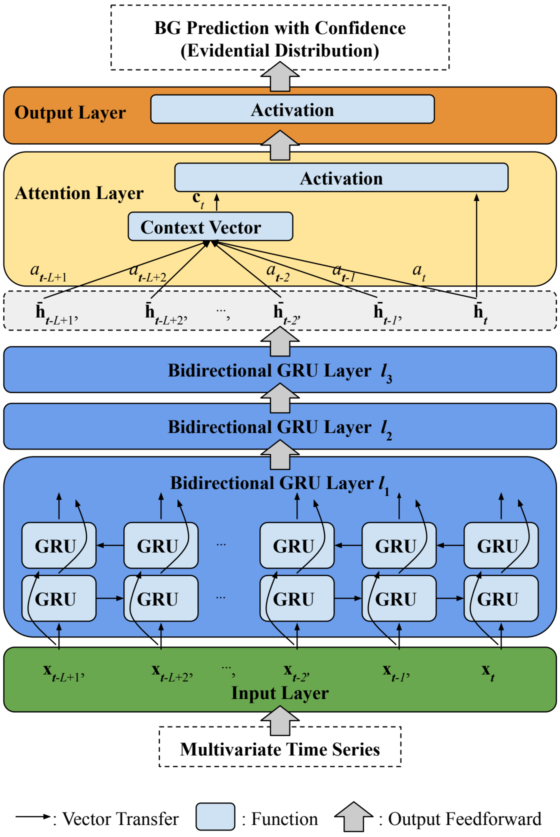

BG Prediction with Confidence
(Evidential Distribution)
Output Layer
Activation
Attention Layer
Activation
$\mathbf{c}_t$
Context Vector
a_{t-L+1}
a_{t-L+2}
a_{t-2}
a_{t-1}
a_t
$\bar{\mathbf{h}}_{t-L+1}$, $\bar{\mathbf{h}}_{t-L+2}$, ..., $\bar{\mathbf{h}}_{t-2}$, $\bar{\mathbf{h}}_{t-1}$, $\bar{\mathbf{h}}_t$
Bidirectional GRU Layer l_3
Bidirectional GRU Layer l_2
Bidirectional GRU Layer l_1
GRU
...
$\mathbf{x}_{t-L+1}$, $\mathbf{x}_{t-L+2}$, ..., $\mathbf{x}_{t-2}$, $\mathbf{x}_{t-1}$, $\mathbf{x}_t$
Input Layer
Multivariate Time Series
: Vector Transfer
: Function
: Output Feedforward

Figure 3.2: Diagram of the proposed deep learning model. The input of multivariate time series is first processed by three bidirectional GRU layers to extract feature maps. The attention layer computes a weighted sum of the hidden states. Then the top layer outputs predictions with confidence from an evidential distribution.

a GRU layer, $\mathcal{H}_{\mathbf{W},\mathbf{U},\mathbf{b}}(\mathbf{x}_t, \mathbf{h}')$ in a direction is given by

$$\begin{aligned} \mathbf{z}_t &= \sigma(\mathbf{W}_z\mathbf{x}_t + \mathbf{U}_z\mathbf{h}' + \mathbf{b}_z), \\ \mathbf{r}_t &= \sigma(\mathbf{W}_r\mathbf{x}_t + \mathbf{U}_r\mathbf{h}' + \mathbf{b}_r), \\ \hat{\mathbf{h}}_t &= \sigma(\mathbf{W}_h\mathbf{x}_t + \mathbf{U}_h\mathbf{r}_t \odot \mathbf{h}' + \mathbf{b}_h), \\ \mathbf{h}_t &= (1 - \mathbf{z}_t) \odot \mathbf{h}' + \mathbf{z}_t \odot \hat{\mathbf{h}}_t, \end{aligned} \tag{3.4}$$

where $\mathbf{z}_t$ and $\mathbf{r}_t$ stand for update gate and reset gate vectors, respectively; σ is the sigmoid function; $\odot$ is the element-wise product; $\mathbf{W}$ and $\mathbf{U}$, and $\mathbf{b}$ are the weights for input, weights for hidden states, and bias, respectively, where the subscripts z, r, and h respectively indicate parameters for the update gate, reset gate, and candidate activation; $\mathbf{h}$, $\mathbf{h}'$, and

$\hat{\mathbf{h}}$ are the hidden states of cell output, cell input and candidate activation, respectively. Concatenating backward and forward output, the state output of a bidirectional GRU layer at timestep t is denoted as $\overline{\mathbf{h}}_t = [\overrightarrow{\mathbf{h}}_t; \overleftarrow{\mathbf{h}}_t]$, where $\overline{\mathbf{h}}$ is a concatenated hidden state in bidirectional RNN, which is given by

$$\begin{aligned} \overrightarrow{\mathbf{h}}_t &= \mathcal{H}_{\overrightarrow{\mathbf{W}},\overrightarrow{\mathbf{U}},\overrightarrow{\mathbf{b}}}(\mathbf{x}_t, \mathbf{h}_{t-1}), \\ \overleftarrow{\mathbf{h}}_t &= \mathcal{H}_{\overleftarrow{\mathbf{W}},\overleftarrow{\mathbf{U}},\overleftarrow{\mathbf{b}}}(\mathbf{x}_t, \mathbf{h}_{t+1}). \end{aligned} \tag{3.5}$$

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

To enable the DNN model to focus on the important parts of the hidden representations, we introduce a many-to-one attention layer at the top of the GRU layers (Figure 3.2). The input of this layer contains a complete sequence of the RNN hidden states. We use a modified general alignment score function [144] to calculate attention weights. Thus, there are a total of two trainable weight matrices ($\mathbf{W}_a$ and $\mathbf{W}_m$) in the attention layer. Each of them is implemented by a dense layer without bias or activation, and therefore it can be updated as a part of the DNN model with gradient descent and back-propagation. The output of the attention layer is an attention vector whose hidden dimension is determined by $\mathbf{W}_m$. Specifically, we first calculate the context vector $\mathbf{c}_t$ as follows

$$\mathbf{c}_t = \sum_{i=t-L+1}^{t} a_i \overline{\mathbf{h}}_i, \tag{3.6}$$

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

$$a_i = \frac{\exp(\texttt{score}(\overline{\mathbf{h}}_i, \overline{\mathbf{h}}_t))}{\sum_i \exp(\texttt{score}(\overline{\mathbf{h}}_i, \overline{\mathbf{h}}_t))}. \tag{3.7}$$

The general alignment score function [144] is formulated as follows.

$$\texttt{score}(\overline{\mathbf{h}}_i, \overline{\mathbf{h}}_t) = \overline{\mathbf{h}}_t^{\top} \mathbf{W}_a \overline{\mathbf{h}}_i. \tag{3.8}$$

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

$$\overline{\mathbf{h}}_t = \tanh(\mathbf{W}_m[\mathbf{c}_t; \overline{\mathbf{h}}_t]), \tag{3.9}$$

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

3.2.3 Evidential Deep Learning

The prediction confidence can be estimated by uncertainty levels. In general, there are two types of uncertainty in deep learning: aleatoric uncertainty, i.e., the uncertainty in the data, and epistemic uncertainty, i.e., the uncertainty in the prediction model [149]. Epistemic uncertainty is crucial to determine out-of-distribution shift, indicating whether the prediction can be trusted or not. Bayesian deep learning is a conventional approach to model epistemic uncertainty, but it heavily relies on a complex sampling process during model training [149]. Therefore, inspired by recent advances on evidential deep learning [150, 153], we employ a process of evidence acquisition to simultaneously model aleatoric uncertainty and epistemic uncertainty. Assuming the targets $y_1, y_2, \ldots, y_t$, are the i.i.d. observations following a Gaussian distribution with unknown mean μ and variance σ^2,

$$\mu \sim \mathcal{N}(\gamma, \sigma^2/\lambda), \quad \sigma^2 \sim \Gamma^{-1}(\alpha, \beta), \tag{3.10}$$

we then have the Gaussian conjugate prior following the normal-inverse-gamma (NIG) distribution: $(\mu, \sigma^2) \sim \texttt{NIG}(\gamma, \lambda, \alpha, \beta)$, as the higher-order evidential distribution. Based on Bayes' theorem, the model evidence $p(y_t|\gamma, \lambda, \alpha, \beta)$, also known as marginal likelihood, can be derived by the likelihood parameters (μ, σ^2) and the probability density function of `NIG`, following a generalised Student-t distribution `St` [150]. In this case, to learn the model parameters by maximum likelihood estimation, the corresponding negative log-likelihood loss function is given by

$$\begin{aligned}\mathcal{L}_t^N &= -\log p(y_t|\gamma,\lambda,\alpha,\beta)\\ &= -\log(\mathtt{St}(y_t|2\alpha,\gamma,\sqrt{\frac{\beta(1+\lambda)}{\lambda\alpha}})),\end{aligned} \tag{3.11}$$

where 2α is the degrees of freedom; γ and $\sqrt{\frac{\beta(1+\lambda)}{\lambda\alpha}}$ are the location parameter and the scale parameter, respectively. Furthermore, a regularizer is introduced to penalise the errors in the prediction [150],

$$\mathcal{L}_t^R = |y_t - \gamma|(2\lambda + \alpha). \tag{3.12}$$

Thus, the loss function on t-th sample is

$$\mathcal{L}_t = \mathcal{L}_t^N + k\mathcal{L}_t^R, \tag{3.13}$$

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

$$\hat{y}_t = \gamma, \quad u_t^a = \sqrt{\frac{\beta}{\alpha - 1}}, \quad u_t^e = \sqrt{\frac{\beta}{\lambda(\alpha - 1)}}. \tag{3.14}$$

3.2.4 Fast Adaptation by Meta-learning

Most commercial CGM sensors measure BG levels at intervals of five minutes. Therefore, a maximum of 288 data points can be collected per day. Training a personalised deep learning model usually requires months of clinical data acquisition. Fortunately, datasets collected from historical clinical trials are available for research purposes. To make use of these datasets and accelerate the training process of new personalised models, meta-learning is a feasible approach. Particularly, we employ MAML [147] with the first-order implementation called Reptile [154]. In terms of BG prediction, the learning tasks in MAML can be referred to as predicting BG levels for different T1D subjects in a cohort

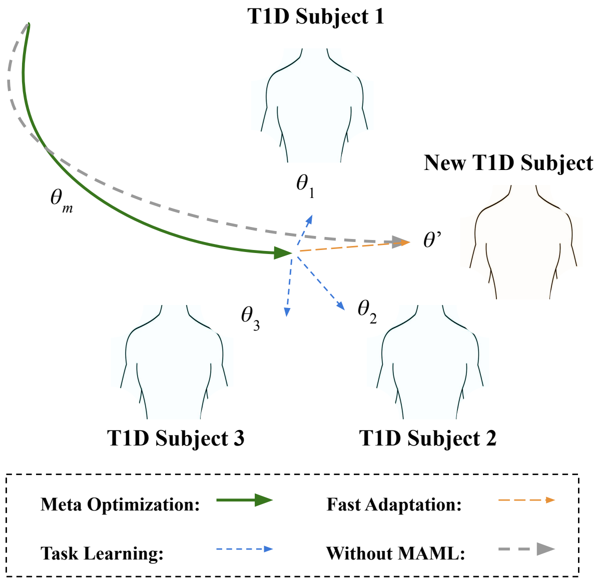


Figure 3.3: Illustration of applying meta-learning to BG prediction for fast adaptation. Meta-learning optimises the model initialization across different subjects in a cohort dataset. Then the initialised model is fast-adaptive to a new subject.

dataset, as shown in Figure 3.3. The meta-models are used in personalised fine-tuning but cannot be used for population-based BG prediction.

The MAML meta-model is expected to minimise loss over a group of different learning tasks (i.e., a T1D cohort). The original MAML relies on an outer loop and an inner loop for meta-optimization and task learning, respectively. Hence, second derivatives are required during the meta-optimization. Calculating a gradient through a gradient is computationally expensive, so a more practical and efficient way to do this is by using the first-order approximation that has the nearly same performance as the original MAML [147]. In this regard, Reptile further simplifies the first-order MAML by reusing the gradients from task learning [154]. Given a task $\mathcal{T}_j$, the j-th update of the model parameters θ_j in the inner loop is defined as

$$\theta_j = \theta_{j-1} - \eta \nabla_{\theta_{j-1}} \mathcal{L}_{B_{\mathcal{T}_j}}(f_{\theta_{j-1}}), \tag{3.15}$$

where η is the learning rate of an Adam optimiser [155]; $\mathcal{L}_{B_{\mathcal{T}_i}}$ stands for the loss of a mini-batch of data samples from the task $\mathcal{T}_i$; and $f_{\theta_{j-1}}$ is the model inference with parameters θ_{j-1}. Then the update of meta-model parameters θ_m across N tasks, i.e., the outer loop, is given by

$$\theta_m \leftarrow \theta_m + \frac{\epsilon}{N} \sum_{j=1}^{N} (\theta_j - \theta_m), \tag{3.16}$$

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

3.3 Experimental Setup

3.3.1 Clinical Datasets

This research uses the following three clinical datasets. It is important to note that, although the number of T1D subjects is not large in each dataset, there is a considerable amount of individual data collected over 6-26 weeks of clinical trials, which is sufficient to develop a personalised deep learning model for each subject. There is normally a 5-10% mean absolute percentage deviation between the measurements of CGM sensors and BG meters. Hence, BG predictions based on CGM might not reflect exact plasma glucose levels but are useful in clinical decision support, because most existing SAP and AP systems automatically adjust the amount of insulin delivery based on CGM measurements and predictions [12, 13, 139]. Recent development in diabetes technology, such as factory-calibrated CGM, has continually reduced sensor error and improved CGM systems [156].

OhioT1DM Dataset

The OhioT1DM dataset is a publicly available dataset [100]. It contains data from 12 subjects with T1D over an eight-week period. All the participants wore a Medtronic Enlite CGM and Medtronic 530G or 630G insulin pumps, and reported their daily events via a smartphone app. Some of them wore Basis Peak or Empatica Embrace bands to collect vital signs data. The data have already been divided into training and testing sets, which account for approximately 80% and 20% of the total samples, respectively.

ARISES Dataset

The ARISES dataset is a proprietary dataset (Imperial College London, London, UK) from a six-week clinical trial (NCT03643692) including 12 T1D participants either on MDI or CSII. Participants in the trial were equipped with Dexcom G6 CGM and logged daily events via mySugr app. In addition, they used Empatica E4 wrist bands and myTracks app. The study was under the protocol (18/LO/1096) approved by London - Fulham Research Ethics Committee in 2018. The details of data collection and clinical study are presented in Section 3.5.1.

ABC4D Dataset

The ABC4D dataset is a proprietary dataset (Imperial College London, London, UK) including data from 25 T1D subjects over six months (NCT02053051) [157]. The dataset contains BG levels measured with Dexcom G5 CGM and multiple self-reported events, including meal ingestion, physical exercise, and basal-bolus insulin regimens. The study was under the protocol (13/LO/0264) approved by London - Chelsea Research Ethics Committee in 2013.

3.3.2 Model Configurations

We performed a two-step hold-out data split (Figure B.1 in Appendix B). The training and testing sets of the OhioT1DM dataset are provided separately. Thus, we first divided the data of each subject in the ARISES and ABC4D datasets into a training set including the first 80% of the data and a hold-out testing set including the remaining 20%. Secondly, we split each training set of the OhioT1DM, ARISES and ABC4D datasets again into an actual training set with the first 80% of the data and a hold-out validation set with the rest 20%. In this way, the training, validation, and testing sets respectively contain 64%, 16%, and 20% of the full data of the ARISES and the ABC4D datasets. The models were trained using the actual training sets, and validation and testing sets were unseen data. Then the validations sets were used for feature selection and hyperparameter tuning. Finally, the testing sets were used to provide unbiased evaluation and the prediction results reported in this work. We strictly followed chronological partitions to split time series data to avoid data leakage and guaranteed that the testing and validation sets did not include any data from the training sets. Similar split methods have been widely used in previous studies on BG prediction [99, 101, 128, 158].

The chosen values for the hyperparameters are listed in Table B.2 in Appendix B. Early stopping was used to mitigate overfitting and improve the generalization of the DNN models, for which we set the total number of epochs to 500 with the patience of 50. We compared the performance of the proposed model with several classic data-driven baseline methods in the literature [19]. SVR [159], random forest regression (RFR) [160], and ARIMA [161] were used as classic machine learning baselines, while bidirectional LSTM (Bi-LSTM) [89] and a variant of the transformer [74] were used as deep learning baselines. The original transformer is a sequence-to-sequence model, so we adopted its encoder as a prediction model [162]. We also used a convolutional recurrent neural network (CRNN) as a baseline method, considering it is the best model in our previous work [96]. We tuned these baseline models using the same hold-out validation sets. The input features of the baseline models are the same as those of FCNN, including CGM, carbohydrate, and bolus insulin, except for ARIMA which only uses CGM data.

We developed the FCNN models and the other DNNs with Python 3.8, TensorFlow 2.3, and Keras 2.4 with a GPU acceleration (NVIDIA GTX 1080 Ti). The classic machine learning methods were developed with scikit-learn 0.23 (SVR, RFR) and statsmodels 0.12 (ARIMA). Finally, we conducted the experiments with the PHs of 30 and 60 minutes. These PHs are commonly employed since they allow timely intervention to prevent undesired glycaemic events [19].

3.3.3 Evaluation Metrics

We used commonly employed evaluation metrics in BG prediction: RMSE, mean absolute error (MAE), and mean absolute percent error (MAPE) with the detailed values of percentages in the A, B, C, D and E regions of CEG [138]. To evaluate the clinical performance, we also used the glucose-specific RMSE (gRMSE) [163] and prediction time delay (PTD) [96, 101, 128] for a comprehensive evaluation. The definition is as follows:

$$\text{RMSE} = \sqrt{\frac{1}{L_T}\sum_{t=1}^{L_T}(G_t - \hat{G}_t)^2}, \tag{3.17}$$

$$\text{MAE} = \frac{1}{L_T}\sum_{t=1}^{L_T}|G_t - \hat{G}_t|, \tag{3.18}$$

$$\text{MAPE} = \frac{1}{L_T}\sum_{t=1}^{L_T}|\frac{G_t - \hat{G}_t}{G_t}|, \tag{3.19}$$

$$\text{gRMSE} = \sqrt{\frac{1}{L_T}\sum_{t=1}^{L_T}P(G_t, \hat{G}_t)(G_t - \hat{G}_t)^2}, \tag{3.20}$$

$$\text{PTD} = \arg\max_k(\sum_{t=1}^{L_T} G_t\hat{G}_{t-k}), \tag{3.21}$$

where L_T denotes the total number of data samples in the testing sets; $P(G_t, \hat{G}_t) = 1 + \alpha_L\bar{\sigma}_{G_t \leq T_L, \beta_L}(G_t)\sigma_{\hat{G}_t \geq G_t, \gamma_L}(\hat{G}_t, G_t) + \alpha_H\sigma_{G_t \geq T_H, \beta_H}(G_t)\bar{\sigma}_{\hat{G}_t \leq G_t, \gamma_H}(\hat{G}_t, G_t)$, $P(G_{t+w}, \hat{G}_{t+w}) \geq 1$, and values of α_L, β_L, γ_L, T_L, α_H, β_H, γ_H, T_H equal to $1.5, 30, 10, 85, 1, 100, 20, 155$.

For classification tasks in hypo- and hyperglycaemia detection, we have a group of standard metrics, including accuracy, sensitivity, precision, specificity, the Matthews correlation coefficient (MCC) (i.e., the Phi coefficient), as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}, \tag{3.22}$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{3.23}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}, \tag{3.24}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{3.25}$$

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{F})(\text{TN} + \text{FN})}}, \tag{3.26}$$

where the number of true positives, true negatives, false positives, and false negatives in a binary classification problem are denoted as TP, TN, FP, and FN. We use additional metrics to indicate alarm fatigue in daily management, including false-positives per day (FPPD) and false alarms per day (FAPD) as follows:

$$\text{FPPD} = \frac{\text{FP}}{\text{Number of Days,}} \tag{3.27}$$

$$\text{FAPD} = \frac{\text{False Hypoglycaemic events}}{\text{Number of Days}}. \tag{3.28}$$

Mean deviation (MD) scores are calculated by the MAE for the glucose sequences in missed predicted hypoglycaemic or hyperglycaemic events.

3.4 Results and Discussion

3.4.1 BG Level Prediction Performance

Table 3.2, 3.3 and 3.4 respectively summarise the prediction results for the OhioT1DM dataset, the ARISES dataset and the ABC4D dataset with PH of 30 and 60 minutes (Mean±STD). Notably, for both PHs, FCNN achieved the best performance on each dataset with the smallest RMSE, MAE, gRMSE, and PTD. Moreover, after evaluating the normality of the results with Shapiro–Wilk tests, we performed paired t-tests to indicate statistical significance with respect to the proposed baseline methods. The FCNN obtained significant improvements on RMSE results for each dataset, but the improvements on gRMSE results are less significant, especially for the ABC4D dataset. The reason for this reduced performance with the ABC4D dataset might be explained by the large number of missing data points in several subjects. The proposed method exhibited good average performance in terms of all the mean errors, but its impact on the clinical benefits needs

Table 3.2: Prediction performance of the considered prediction methods on the OhioT1DM dataset

Methods	RMSE (mg/dL) ↓	MAE (mg/dL) ↓	gRMSE (mg/dL) ↓	PTD (min) ↓
PH = 30 minutes				
FCNN	$\mathbf{18.64 \pm 2.60}$	$\mathbf{13.25 \pm 1.67}$	$\mathbf{22.86 \pm 3.47}$	$\mathbf{6.75 \pm 4.30}$
CRNN [96]	$19.38 \pm 2.39^{\dagger}$	$13.79 \pm 1.64^{*}$	$24.12 \pm 3.14^{\dagger}$	7.42 ± 4.66
Bi-LSTM [89]	$19.59 \pm 2.22^{\ddagger}$	$13.79 \pm 1.31^{\dagger}$	$24.30 \pm 3.14^{\ddagger}$	6.82 ± 4.08
Transformer [74]	$19.69 \pm 2.36^{\ddagger}$	$13.90 \pm 1.42^{\ddagger}$	$24.60 \pm 3.25^{\ddagger}$	7.56 ± 4.49
SVR [159]	$21.10 \pm 2.31^{\ddagger}$	$15.98 \pm 1.94^{\ddagger}$	$26.49 \pm 3.14^{\ddagger}$	$7.94 \pm 3.76^{*}$
RFR [160]	$21.18 \pm 2.26^{\ddagger}$	$15.30 \pm 1.61^{\ddagger}$	$26.33 \pm 2.97^{\ddagger}$	$8.25 \pm 3.86^{*}$
ARIMA [161]	$20.39 \pm 2.21^{\ddagger}$	$14.40 \pm 1.41^{\ddagger}$	$24.48 \pm 2.68^{\ddagger}$	$11.36 \pm 4.43^{\ddagger}$
PH = 60 minutes				
FCNN	$\mathbf{31.07 \pm 3.62}$	$\mathbf{22.86 \pm 2.89}$	$\mathbf{39.78 \pm 5.28}$	$\mathbf{14.58 \pm 9.91}$
CRNN [96]	$32.02 \pm 3.76^{\ddagger}$	$23.82 \pm 3.13^{\ddagger}$	$41.25 \pm 5.37^{\ddagger}$	16.01 ± 10.03
Bi-LSTM [89]	$33.44 \pm 3.76^{\ddagger}$	$24.59 \pm 2.89^{\ddagger}$	$43.45 \pm 5.42^{\ddagger}$	16.71 ± 8.82
Transformer [74]	$32.96 \pm 3.70^{\ddagger}$	$24.19 \pm 2.79^{\ddagger}$	$42.82 \pm 5.22^{\dagger}$	14.81 ± 10.66
SVR [159]	$33.83 \pm 3.62^{\ddagger}$	$25.63 \pm 2.98^{\ddagger}$	$43.88 \pm 5.07^{\ddagger}$	$24.00 \pm 9.67^{\ddagger}$
RFR [160]	$35.31 \pm 3.72^{\ddagger}$	$26.43 \pm 3.02^{\ddagger}$	$45.32 \pm 5.13^{\ddagger}$	$23.10 \pm 10.61^{\dagger}$
ARIMA [161]	$35.42 \pm 3.74^{\ddagger}$	$25.97 \pm 2.70^{\ddagger}$	$43.78 \pm 4.68^{\ddagger}$	$35.12 \pm 10.58^{\ddagger}$

	CEG-Regions (%)				
Methods	A ↑	B ↑	C ↓	D ↓	E ↓
PH = 30 minutes					
FCNN	$\mathbf{89.80 \pm 3.65}$	8.96 ± 2.85	0.01 ± 0.02	1.22 ± 0.93	0.01 ± 0.01
CRNN [96]	$88.72 \pm 3.90^{*}$	$9.72 \pm 2.98^{*}$	0.02 ± 0.03	$1.55 \pm 1.14^{*}$	0.00 ± 0.00
Bi-LSTM [89]	$88.93 \pm 3.07^{*}$	$9.74 \pm 2.49^{\ddagger}$	0.02 ± 0.03	1.30 ± 0.84	0.01 ± 0.02
Transformer [74]	$89.15 \pm 3.46^{*}$	$9.77 \pm 2.80^{\ddagger}$	0.01 ± 0.03	1.07 ± 0.71	0.00 ± 0.01
SVR [159]	$82.70 \pm 6.05^{\ddagger}$	$14.58 \pm 4.32^{\ddagger}$	0.01 ± 0.02	$2.71 \pm 2.23^{*}$	0.00 ± 0.01
RFR [160]	$86.71 \pm 4.48^{\ddagger}$	$11.70 \pm 3.52^{\ddagger}$	0.02 ± 0.04	$1.56 \pm 1.10^{*}$	0.00 ± 0.01
ARIMA [161]	$88.41 \pm 3.48^{\ddagger}$	$10.94 \pm 3.19^{\ddagger}$	0.03 ± 0.06	$0.61 \pm 0.38^{*}$	0.01 ± 0.01
PH = 60 minutes					
FCNN	$\mathbf{72.58 \pm 7.87}$	24.39 ± 6.41	0.16 ± 0.14	2.85 ± 1.68	0.02 ± 0.04
CRNN [96]	$71.06 \pm 8.69^{*}$	25.57 ± 7.07	0.15 ± 0.17	$3.20 \pm 1.99^{*}$	0.01 ± 0.04
Bi-LSTM [89]	$70.61 \pm 8.21^{\dagger}$	$25.98 \pm 6.70^{*}$	0.17 ± 0.13	3.19 ± 1.89	0.05 ± 0.07
Transformer [74]	71.70 ± 7.77	25.20 ± 6.43	0.15 ± 0.15	2.92 ± 1.65	0.04 ± 0.05
SVR [159]	$66.43 \pm 9.15^{\ddagger}$	$29.61 \pm 7.30^{\dagger}$	0.20 ± 0.21	$3.73 \pm 2.62^{*}$	0.03 ± 0.04
RFR [160]	$67.03 \pm 8.17^{\ddagger}$	$29.38 \pm 6.29^{\ddagger}$	0.23 ± 0.19	$3.34 \pm 2.14^{*}$	0.02 ± 0.04
ARIMA [161]	$68.77 \pm 6.85^{*}$	$28.65 \pm 5.83^{\ddagger}$	$0.46 \pm 0.40^{*}$	$2.06 \pm 1.00^{*}$	0.05 ± 0.05

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

to be further improved. This is the reason why we introduced model confidence to predict adverse glycaemic events with adjustable lower and upper bounds (Section 3.4.2).

Overall, the accuracy of BG prediction decreased as the PHs became larger. The

Table 3.3: Prediction performance of the considered prediction methods on the ARISES dataset

Methods	RMSE (mg/dL) ↓	MAE (mg/dL) ↓	gRMSE (mg/dL) ↓	PTD (min) ↓
		PH = 30 minutes		
FCNN	**20.23 ± 3.38**	**14.67 ± 2.36**	**25.20 ± 4.52**	**8.36 ± 4.22**
CRNN [96]	$20.76 \pm 3.71^{\ddagger}$	$15.02 \pm 2.62^{*}$	$26.14 \pm 4.98^{\dagger}$	8.64 ± 4.73
Bi-LSTM [89]	$20.95 \pm 3.11^{\ddagger}$	$15.29 \pm 2.22^{\ddagger}$	$26.54 \pm 4.13^{*}$	8.49 ± 4.48
Transformer [74]	$22.45 \pm 4.08^{\ddagger}$	$16.38 \pm 3.01^{\ddagger}$	$28.73 \pm 5.54^{\ddagger}$	$10.24 \pm 5.36^{\ddagger}$
SVR [159]	$22.26 \pm 4.21^{\ddagger}$	$16.85 \pm 3.20^{\ddagger}$	$28.19 \pm 5.85^{\ddagger}$	9.52 ± 4.82
RFR [160]	$23.86 \pm 4.44^{\ddagger}$	$17.56 \pm 3.18^{\ddagger}$	$30.14 \pm 5.99^{\ddagger}$	$10.28 \pm 5.21^{\ddagger}$
ARIMA [161]	$21.75 \pm 4.02^{\ddagger}$	$15.59 \pm 2.70^{\ddagger}$	$26.21 \pm 5.23^{*}$	$11.56 \pm 4.89^{\ddagger}$
		PH = 60 minutes		
FCNN	**35.40 ± 7.04**	**26.23 ± 5.00**	**45.93 ± 9.60**	**24.44 ± 11.05**
CRNN [96]	$36.08 \pm 6.99^{*}$	$26.86 \pm 5.14^{\ddagger}$	$47.58 \pm 10.03^{\ddagger}$	24.84 ± 11.24
Bi-LSTM [89]	$36.83 \pm 7.27^{\ddagger}$	$27.44 \pm 5.38^{\ddagger}$	48.13 ± 10.09‡	25.53 ± 11.48
Transformer [74]	$36.98 \pm 6.96^{\ddagger}$	$27.59 \pm 5.27^{\ddagger}$	$48.66 \pm 10.01^{\ddagger}$	26.14 ± 12.30
SVR [159]	$37.06 \pm 7.55^{\ddagger}$	$27.94 \pm 5.66^{\ddagger}$	$48.74 \pm 10.71^{\ddagger}$	27.32 ± 12.17
RFR [160]	$39.46 \pm 7.73^{\ddagger}$	$29.78 \pm 5.87^{\ddagger}$	$51.58 \pm 10.79^{\ddagger}$	26.69 ± 11.91
ARIMA [161]	$39.46 \pm 8.13^{\ddagger}$	$28.71 \pm 5.62^{\ddagger}$	$49.70 \pm 10.97^{\ddagger}$	$36.00 \pm 12.44^{\ddagger}$

			CEG-Regions (%)		
Methods	A ↑	B ↑	C ↓	D ↓	E ↓
			PH = 30 minutes		
FCNN	**87.24 ± 5.86**	10.60 ± 4.20	0.02 ± 0.07	2.14 ± 1.75	0.00 ± 0.00
CRNN [96]	86.92 ± 6.12	10.79 ± 4.48	0.03 ± 0.05	2.27 ± 1.81	0.00 ± 0.00
Bi-LSTM [89]	86.63 ± 6.21	10.92 ± 4.38	0.02 ± 0.04	2.43 ± 1.92	0.00 ± 0.00
Transformer [74]	$85.29 \pm 7.50^{*}$	$12.37 \pm 5.53^{\ddagger}$	0.04 ± 0.09	2.30 ± 2.03	0.00 ± 0.00
SVR	$82.57 \pm 8.57^{\ddagger}$	$14.45 \pm 6.37^{\ddagger}$	0.03 ± 0.05	2.94 ± 2.35	0.00 ± 0.00
RFR [160]	$83.22 \pm 7.19^{\ddagger}$	$14.24 \pm 5.44^{\ddagger}$	0.07 ± 0.11	2.47 ± 1.85	0.00 ± 0.00
ARIMA [161]	86.43 ± 6.07	$12.68 \pm 5.42^{\ddagger}$	0.04 ± 0.06	$0.85 \pm 0.72^{\ddagger}$	0.00 ± 0.00
			PH = 60 minutes		
FCNN	**69.29 ± 8.91**	26.27 ± 6.51	0.33 ± 0.46	4.11 ± 2.60	0.01 ± 0.02
CRNN [96]	68.52 ± 9.56	26.99 ± 7.15	0.24 ± 0.28	4.24 ± 2.63	0.01 ± 0.03
Bi-LSTM [89]	$67.50 \pm 9.51^{\ddagger}$	$28.00 \pm 7.09^{\ddagger}$	0.30 ± 0.39	4.18 ± 2.61	0.01 ± 0.03
Transformer [74]	$67.14 \pm 9.57^{\ddagger}$	$28.33 \pm 7.11^{\ddagger}$	0.21 ± 0.29	4.30 ± 2.66	0.02 ± 0.04
SVR [159]	$66.42 \pm 10.32^{\dagger}$	$28.94 \pm 7.88^{*}$	0.31 ± 0.38	4.30 ± 2.80	0.03 ± 0.06
RFR [160]	$64.34 \pm 9.51^{\ddagger}$	$30.92 \pm 7.20^{\ddagger}$	0.41 ± 0.43	4.24 ± 2.54	$0.09 \pm 0.12^{*}$
ARIMA [161]	$67.05 \pm 9.52^{\ddagger}$	$29.19 \pm 7.76^{\ddagger}$	$0.56 \pm 0.53^{\ddagger}$	$2.91 \pm 1.65^{\dagger}$	$0.29 \pm 0.34^{*}$

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

daily events that might occur within the 60-minute period (e.g. meals, corrections boluses, exercise) have an affect on BG levels and make the prediction more challenging. It is worth noting that all the three deep learning methods outperformed the classic machine learning baselines, demonstrating the good learning behaviors of DNNs. Regarding the

Table 3.4: Prediction performance of the considered prediction methods on the ABC4D dataset

Methods	RMSE (mg/dL) ↓	MAE (mg/dL) ↓	gRMSE (mg/dL) ↓	PTD (min) ↓
PH = 30 minutes				
FCNN	$\mathbf{20.25 \pm 2.60}$	$\mathbf{14.50 \pm 1.95}$	$\mathbf{25.00 \pm 3.49}$	$\mathbf{7.76 \pm 3.79}$
CRNN [96]	$20.55 \pm 2.55^{*}$	$14.77 \pm 1.86^{\dagger}$	$25.44 \pm 3.46^{*}$	8.23 ± 4.62
Bi-LSTM [89]	$20.66 \pm 2.45^{\ddagger}$	$14.85 \pm 1.77^{\ddagger}$	$25.68 \pm 3.22^{\ddagger}$	7.90 ± 4.42
Transformer [74]	$20.63 \pm 2.66^{\dagger}$	$14.84 \pm 1.91^{\ddagger}$	$25.60 \pm 3.59^{\ddagger}$	8.87 ± 4.85
SVR [159]	$21.90 \pm 2.44^{\ddagger}$	$16.73 \pm 1.85^{\ddagger}$	$27.70 \pm 3.45^{\ddagger}$	8.71 ± 4.65
RFR [160]	$21.80 \pm 2.48^{\ddagger}$	$15.83 \pm 1.81^{\ddagger}$	$27.12 \pm 3.34^{\ddagger}$	8.69 ± 4.63
ARIMA [161]	$22.13 \pm 2.60^{\ddagger}$	$15.60 \pm 1.90^{\ddagger}$	$26.48 \pm 3.56^{\ddagger}$	$13.55 \pm 4.65^{\ddagger}$
PH = 60 minutes				
FCNN	$\mathbf{34.03 \pm 4.74}$	$\mathbf{25.26 \pm 3.60}$	$\mathbf{44.06 \pm 6.25}$	$\mathbf{21.47 \pm 7.59}$
CRNN [96]	$34.39 \pm 4.75^{\ddagger}$	$25.63 \pm 3.53^{\ddagger}$	$44.58 \pm 6.31^{\dagger}$	$25.56 \pm 10.94^{\dagger}$
Bi-LSTM [89]	$34.82 \pm 4.44^{\dagger}$	$25.86 \pm 3.27^{*}$	$45.17 \pm 5.98^{*}$	22.78 ± 9.78
Transformer [74]	$34.91 \pm 4.43^{\dagger}$	$25.99 \pm 3.28^{*}$	$45.42 \pm 6.13^{\dagger}$	$26.08 \pm 11.65^{*}$
SVR [159]	$35.37 \pm 4.70^{\ddagger}$	$26.98 \pm 3.50^{\ddagger}$	$45.99 \pm 6.43^{\ddagger}$	$26.99 \pm 11.30^{\ddagger}$
RFR [160]	$36.57 \pm 4.76^{\ddagger}$	$27.43 \pm 3.55^{\ddagger}$	$47.27 \pm 6.40^{\ddagger}$	$24.95 \pm 10.28^{*}$
ARIMA [161]	$38.55 \pm 5.15^{\ddagger}$	$28.00 \pm 3.75^{\ddagger}$	$47.94 \pm 7.16^{\ddagger}$	$39.42 \pm 7.48^{\ddagger}$

	CEG-Regions (%)				
Methods	A ↑	B ↑	C ↓	D ↓	E ↓
PH = 30 minutes					
FCNN	$\mathbf{86.50 \pm 3.90}$	11.54 ± 2.84	0.03 ± 0.06	1.93 ± 1.20	0.00 ± 0.01
CRNN [96]	$85.94 \pm 3.75^{\dagger}$	11.78 ± 2.46	0.03 ± 0.06	$2.25 \pm 1.47^{\ddagger}$	0.00 ± 0.00
Bi-LSTM [89]	$85.72 \pm 4.15^{\dagger}$	11.81 ± 2.74	0.03 ± 0.06	$2.43 \pm 1.72^{\ddagger}$	0.00 ± 0.02
Transformer [74]	$85.89 \pm 3.98^{*}$	11.92 ± 2.68	0.03 ± 0.06	2.17 ± 1.56	0.00 ± 0.00
SVR [159]	$79.87 \pm 6.55^{\ddagger}$	$15.59 \pm 3.28^{\ddagger}$	0.03 ± 0.06	$4.51 \pm 4.30^{\ddagger}$	0.00 ± 0.00
RFR [160]	$83.91 \pm 4.30^{\ddagger}$	$13.37 \pm 2.65^{\ddagger}$	0.03 ± 0.06	$2.68 \pm 1.93^{\ddagger}$	0.00 ± 0.01
ARIMA [161]	$85.11 \pm 3.81^{\ddagger}$	$13.67 \pm 3.18^{\ddagger}$	$0.06 \pm 0.06^{\ddagger}$	$1.14 \pm 0.77^{\ddagger}$	$0.01 \pm 0.02^{\ddagger}$
PH = 60 minutes					
FCNN	$\mathbf{68.01 \pm 5.37}$	27.07 ± 3.47	0.23 ± 0.24	4.67 ± 2.87	0.03 ± 0.04
CRNN [96]	$67.07 \pm 5.53^{\ddagger}$	$27.72 \pm 3.29^{*}$	0.24 ± 0.29	4.95 ± 3.35	0.02 ± 0.05
Bi-LSTM [89]	$66.71 \pm 5.89^{*}$	$28.13 \pm 3.56^{*}$	0.25 ± 0.25	4.88 ± 3.35	0.03 ± 0.05
Transformer [74]	$66.68 \pm 5.89^{*}$	28.26 ± 3.88	0.22 ± 0.26	4.82 ± 3.12	0.02 ± 0.04
SVR [159]	$63.62 \pm 7.01^{\ddagger}$	$30.44 \pm 4.04^{\ddagger}$	$0.28 \pm 0.29^{*}$	$5.62 \pm 4.54^{*}$	0.03 ± 0.05
RFR [160]	$64.60 \pm 6.22^{\ddagger}$	$30.02 \pm 3.76^{\ddagger}$	$0.36 \pm 0.36^{\ddagger}$	4.98 ± 3.63	0.04 ± 0.06
ARIMA [161]	$65.22 \pm 4.88^{\ddagger}$	$30.94 \pm 3.63^{\ddagger}$	$0.59 \pm 0.46^{\ddagger}$	$3.05 \pm 1.55^{\ddagger}$	$0.20 \pm 0.15^{\ddagger}$

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

RMSE performance, the transformer is comparable to the Bi-LSTM, which exhibits better results with the 60-minute PH in the OhioT1DM dataset, and with the 30-minute PH in the ABC4D dataset.

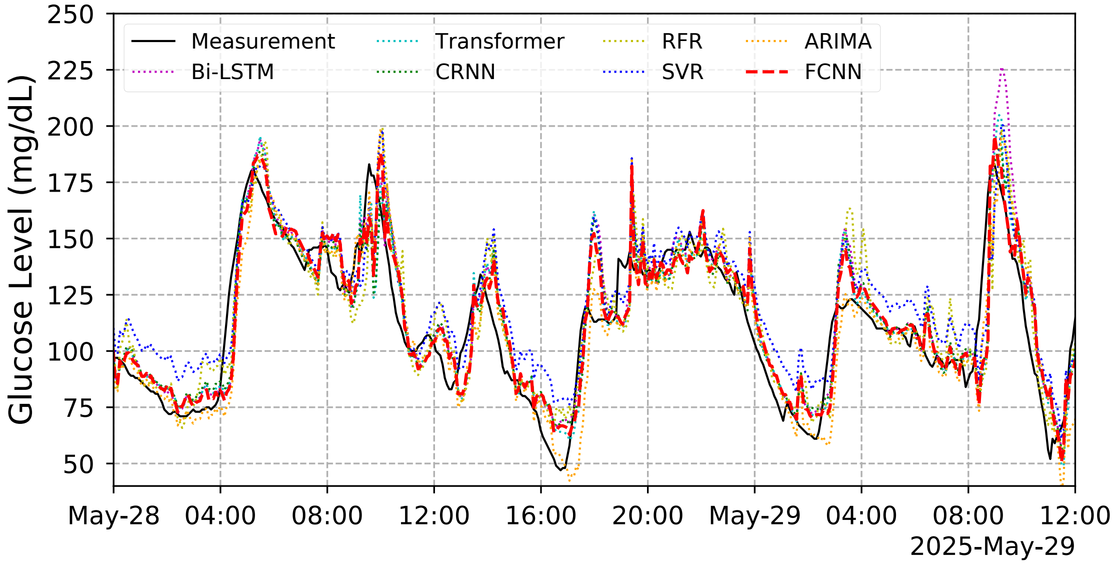


Figure 3.4: 1.5-day period forecasting performance of the considered methods over the 30-minute PH for the OhioT1DM dataset. The solid black line indicates BG levels measured by CGM, and the dashed red line indicates the prediction results of FCNN. The magenta, cyan, green, yellow, blue, and orange lines respectively indicate the baselines of Bi-LSTM, Transformer, CRNN, RFR, SVR, and ARIMA.

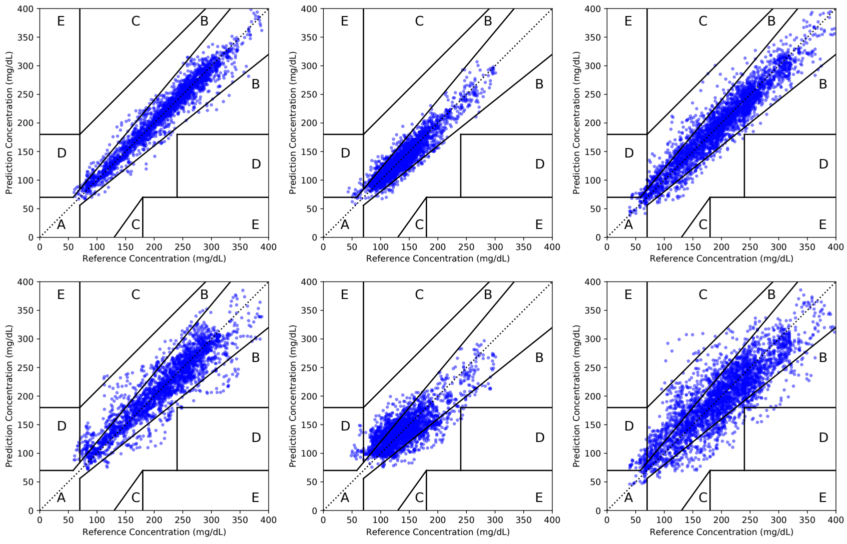


Figure 3.5: CEG analysis over 30-minute (top) and 60-minute (bottom) PHs for three T1D subjects from the OhioT1DM, ARISES, and ABC4D datasets (from left to right), receptively.

Figure 3.4 shows the trajectories of the BG predictions on a subject in the OhioT1DM dataset. Compared with the baseline methods, we see that the FCNN performed well, especially at the peaks and troughs of the BG curve. That is, the errors in the hypoglycaemia and hyperglycaemia regions are small (e.g., the predictions around 9:00 on May 29). Similar performance can be observed in the ABC4D dataset and the ARISES dataset. These findings on the BG trajectories are consistent with the results in the above tables. Figure 3.5 depicts the CEG analysis for FCNN, which is evaluated on three T1D subjects and two PHs. We observed that the dots concentrate on the A and B regions. These two regions include the predictions that are within 20% error with respect to the actual CGM measurements, and would not lead to inappropriate treatment. The corresponding numerical results of each region are presented in Table 3.2, 3.3 and 3.4. It is to be noted that, when evaluated on the OhioT1DM dataset, 98.86% of FCNN predictions for the 30-minute PH are located within the A and B regions.

3.4.2 Confidence for Hypoglycaemia Prediction

Although FCNN achieves substantial improvements in these regions, a significant delay can still be observed in Figure 3.4. To address this clinical challenge, we incorporated the evidential deep learning and modeled uncertainty defined in Equation (3.14). Figure 3.6 depicts an instance of using the lower bounds of the CI to successfully identify two hypoglycaemia regions in the dashed black ellipses, which are likely to be missed using single prediction values. Considering epistemic uncertainty interprets the confidence of predictions, we calculated upper and lower bounds as $[\hat{G}_{t+r} - f_N^{-1}(u_t^e), \hat{G}_{t+r} + f_N^{-1}(u_t^e)]$.

Hypoglycaemia is more dangerous than hyperglycaemia, since it can lead to acute coma or even death in severe cases [25]. To quantify the effectiveness of the prediction confidence, we altered the bounds by $[\hat{G}_{t+r} - zf_N^{-1}(u_t^e), \hat{G}_{t+r} + zf_N^{-1}(u_t^e)]$, where z is the ratio of the uncertainty and treated as a hyperparameter with a range of [0, 1]. We evaluated the results using four classification metrics: sensitivity, precision, FPPD, and MCC. Considering that hypoglycaemia is a minority class, accounting for 3-5% of whole BG trajectories (Table B.4 in Appendix B), we use FPPD as an alternative way to present the specificity. It indicates how often the algorithm would lead to false hypoglycaemia alarms in decision support systems or closed-loop systems. Rescue treatment for impending hy-

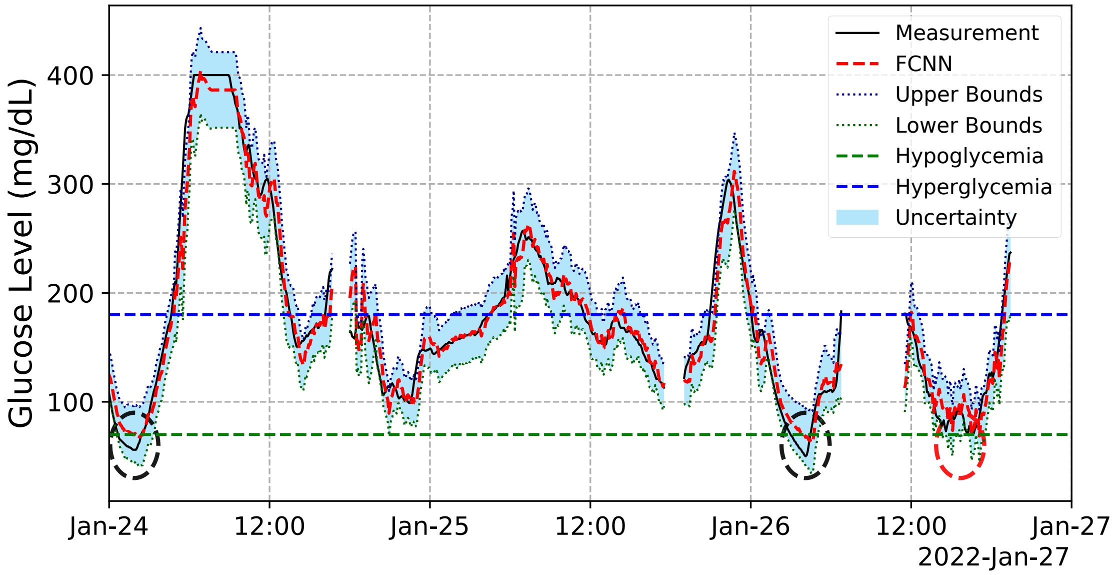


Figure 3.6: Forecasting results corresponding to the proposed method including prediction confidence on a subject in the OhioT1DM dataset. The solid black line indicates CGM measurements, and the dashed red line indicates the prediction results of FCNN. The dotted blue and green lines respectively indicate the upper and lower bounds derived by evidential regression. The shaded light blue area indicates the CI for upper and lower bounds. The dashed blue and green lines respectively indicate the thresholds of hyperglycaemia and hypoglycaemia.

poglycaemia, such as rescue carbohydrates, is commonly performed in T1D management. Hence, a hypoglycaemic event was considered when there was a single BG level below 70 mg/dL. Table 3.5 presents the hypoglycaemia prediction performance of FCNN and the baseline methods. It is noted that, FCNN outperformed all the considered baseline methods with higher MCC scores for the 30-minute and 60-minute PHs. Although ARIMA exhibited the large RMSE and PTD results for BG prediction across the three datasets, it obtained good performance for hypoglycaemia prediction. A possible explanation is that ARIMA performed well at the troughs of BG curves but showed significant delay in other regions (Figure 3.4). It is reasonable that FCNN yields high FPPD due to the use of the lower bounds for detecting hypoglycaemia. However, most false positives occurred around actual hypoglycaemic events. If each event triggers a single alarm, FCNN can still achieve a small number of FAPD of 0.48 ± 0.53 for the 60-minute PH.

There is a trade-off between precision and sensitivity, while high MCC scores can

Table 3.5: Performance of hypoglycaemia prediction methods on the OhioT1DM dataset

Method	Sensitivity (%) ↑	Precision (%) ↑	FPPD ↓	MCC ↑
		PH = 30 minutes		
FCNN	84.09 ± 5.73	65.60 ± 17.68	6.07 ± 4.30	0.72 ± 0.10
CRNN [96]	$44.34 \pm 25.53^{\ddagger}$	70.03 ± 23.33	1.88 ± 1.43	$0.53 \pm 0.23^{\ddagger}$
Bi-LSTM [89]	$45.12 \pm 23.13^{\ddagger}$	56.58 ± 19.93	4.71 ± 3.50	$0.48 \pm 0.20^{\ddagger}$
Transformer [74]	$59.35 \pm 18.82^{\ddagger}$	71.36 ± 16.48	3.69 ± 1.94	$0.61 \pm 0.15^{\ddagger}$
SVR [159]	$12.84 \pm 11.73^{\ddagger}$	54.84 ± 36.37	$1.12 \pm 1.04^{\ddagger}$	$0.24 \pm 0.18^{\ddagger}$
RFR [160]	$47.86 \pm 25.16^{\ddagger}$	62.49 ± 21.08	3.21 ± 2.21	$0.52 \pm 0.21^{\ddagger}$
ARIMA [161]	86.32 ± 6.04	$50.94 \pm 9.61^{*}$	10.87 ± 4.74	0.63 ± 0.06
		PH = 60 minutes		
FCNN	68.58 ± 14.39	60.64 ± 18.82	14.45 ± 15.86	0.59 ± 0.08
CRNN [96]	$15.83 \pm 13.30^{\ddagger}$	42.58 ± 29.52	2.60 ± 2.27	$0.23 \pm 0.18^{\ddagger}$
Bi-LSTM [89]	$28.73 \pm 19.32^{\ddagger}$	52.77 ± 26.34	5.77 ± 3.78	$0.34 \pm 0.17^{\ddagger}$
Transformer [74]	$38.71 \pm 27.98^{*}$	53.92 ± 29.71	5.81 ± 4.94	$0.40 \pm 0.23^{*}$
SVR [159]	$9.46 \pm 8.91^{\ddagger}$	$34.29 \pm 30.45^{*}$	$3.25 \pm 2.95^{*}$	$0.15 \pm 0.14^{\ddagger}$
RFR [160]	$19.03 \pm 24.44^{\ddagger}$	32.63 ± 30.99	2.70 ± 3.02	$0.22 \pm 0.19^{\ddagger}$
ARIMA [161]	80.17 ± 8.99	43.80 ± 9.51	19.55 ± 9.33	0.54 ± 0.07

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

be obtained only if model performs well on all the confusion matrix categories. To avoid hypoglycaemia, higher sensitivity is preferred in clinical settings at the cost of slightly less precision (i.e., increase of false positive rate), as indicated by the dashed black and red ellipses in Figure 3.6, but too low precision might cause alarm fatigue. In this regard, the ratio z is flexible and can be chosen by clinicians.

3.4.3 Adaptation Performance With Limited Data

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

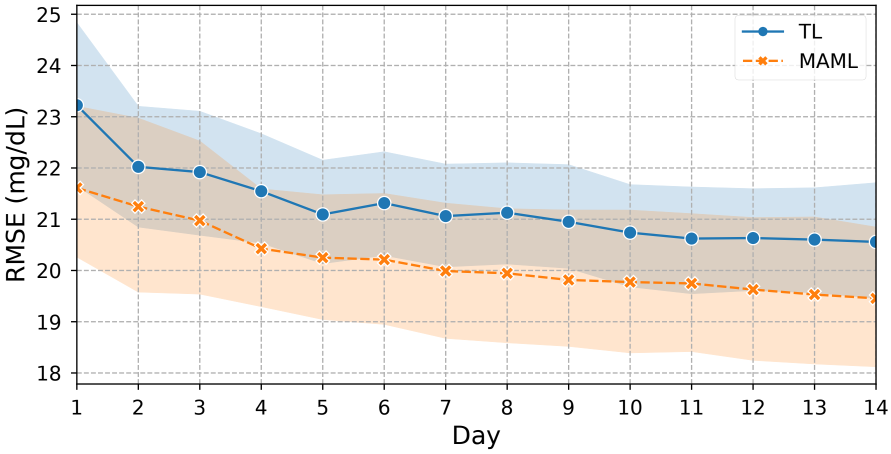


Figure 3.7: 30-minute RMSE results of fine-tuning on the FCNN meta-model and the TL-pretrained model with the small-size training sets of the OhioT1DM dataset. The solid blue and orange lines are the mean RMSE of TL and MAML, respectively, and shaded areas stand for 95% CI.

For comparison purposes, a pretrained model with TL techniques was employed as a baseline, which has been commonly used in the literature of BG prediction [99, 128, 146, 114]. The TL-pretrained model has the same DNN architecture as that of FCNN, and developing such a model includes two steps. First, we excluded the data of the target subject and combined the training data of the remaining subjects to form a global set. Then, we pretrained a model with the mini-batch data randomly sampled from the global set, and each input batch corresponds to a single subject.

After performing MAML and TL without the hold-out data, we fine-tuned the meta-model and TL-pretrained model using the same individual data of the hold-out target subject. In order to simulate the 14-day use case across a lifespan of a CGM sensor, we sequentially added one-day data into the individual data set and repeated the fine-tuning experiments on each experimental day. Each time we used the same testing sets as those in the previous Section 3.4.1 for evaluation. The experiments were repeated for all the subjects.

Figure 3.7 depicts the performance of FCNN and the TL baseline on the OhioT1DM dataset. Table 3.6 summarises the results for the three datasets. In this limit case, when compared with TL, MAML provided an average RMSE improvement of 1.48 and 0.96

Table 3.6: 30-minute RMSE of the fast adaptation methods across the three datasets

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

mg/dL on day 1 and day 14, respectively. The meta-models by MAML exhibited better prediction performance during the whole fine-tuning process and achieved much smaller RMSE from the start of the fine-tuning (day 1), when the size of available data was extremely small. These findings are consistent with a recent study on MAML, which has proven that the effectiveness of MAML is primarily due to feature reuse [164]. That is, the meta-initialised models were already good at learning representations for a new subject. MAML achieved an RMSE below 20 on day 7, while it took much longer for the TL-pretrained models to reach this level. We further performed the experiments using the data of the first 25 days, which is the maximum mutual length of the data in the training sets. It is observed that the RMSE of MAML keep decreasing until day 18 for the OhioT1DM and ARISES datasets, and day 20 for the ABC4D dataset, and then become stable.

In the general case, where enough training data were available for each subject, the use of MAML significantly reduced RMSE by 0.2 mg/dL ($p < 0.005$, Figure 3.8). These results indicate that the use of MAML is a feasible approach to enable fast adaptation and improve model performance with a small size of available data. It should be noted that the MAML meta-models are not the population models for BG prediction, which cannot be directly evaluated on the testing data of new subjects. Here we only indicate that MAML outperformed TL during the fine-tuning of the development of personalised models.

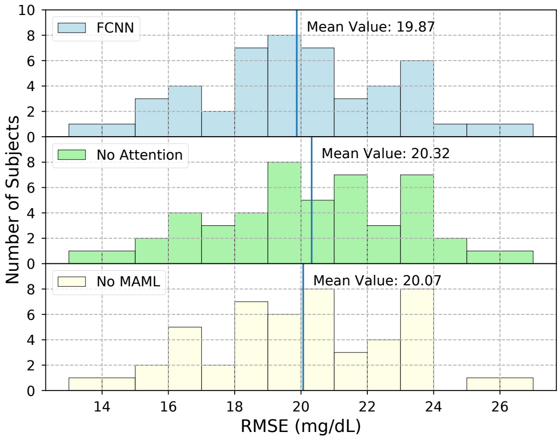


Figure 3.8: Histogram of the RMSE performance across the three datasets in ablation analysis. Each bin stands for the number of subjects within the corresponding RMSE range.

3.4.4 Comparison With Existing Studies

There are several significant differences between the attention-based LSTM model presented by Mirshekarian *et al.* [99] and the proposed model in this chapter. Firstly, the authors evaluated the model on a previous version of the OhioT1DM dataset (2018 version), which contains six of the 12 T1D subjects in the 2020 OhioT1DM dataset that we used. If we evaluate our models on the six 2018 subjects, a 30-minute RMSE of 18.10 mg/dL can be obtained, which is lower than the best result of 18.70 mg/dL that the authors reported using the same experimental settings (i.e., agnostic scenarios without what-if events). Secondly, the authors used an additive form of attention module [141] to obtain a hidden state of LSTM cells with the largest attention weight, while we use a general form of attention mechanism [144] to calculate the weighted sum of the hidden states of bidirectional GRU cells. Finally, their attention module did not improve BG prediction for real clinical data, while ours significantly reduced RMSE by 0.45 mg/dL ($p < 0.005$) in ablation analysis (Figure 3.8). It is to be noted that, in the literature, the sensitivity of 75% and precision of 51% [165], and the sensitivity of 59% and precision of 68% [114], were obtained for a 30-minute PH, while an MCC score of 0.51 with the sensitivity of 48.5% was achieved for a 60-minute PH [166]. Therefore, the performance of FCNN (Table 3.5) is better than the state of the art. A potential improvement can be made by introducing another regularizer into the loss function to penalise the error of hypoglycaemia detection (Equation (3.13)), although it may result in a larger RMSE for BG prediction.

In the experiments, we have shown that FCNN is a superior prediction method when compared with the chosen baseline algorithms. In general, it is difficult to perform a fair comparison with the existing work, due to unavailable code, data, and experimental settings. A recent work proposed the glucose variability impact index and glucose prediction consistency index as a method to assess the correlation between RMSE results of BG prediction and glucose variability [167], which can be used to compare algorithms across different studies. Here we measured glucose variability by means of the coefficient of variation (CV) and applied linear least-squares regression to obtain indices for each prediction method. However, it is to be noted that the correlation results on the OhioT1DM and ARISES dataset are not significant, possibly due to the small numbers of T1D subjects. Thus, we reported the results with Pearson correlation coefficients (r) and p-values (p) on the ABC4D dataset in Table B.5 of Appendix B. It is worth noting that the FCNN method achieved small indices for both PHs, indicating that glucose variability has a low impact on the accuracy and consistency of BG prediction. Moreover, the FCNN framework can be adapted to many existing DNN models to improve their performance, such as the CNN [101], the CRNN [96] and the dilated RNN [128]. Particularly, such adaptation only involves three steps: replacing the dense top layer with evidential output layer, using the corresponding negative log-likelihood loss, and applying the MAML procedures.

To further improve the accuracy of BG prediction, it is viable to incorporate physical activity data, since aerobic exercise would cause rapid changes in BG concentration and has significant impacts on the action of insulin and other hormones [168]. Thus, real-time BG prediction is critical for physically active people with T1D, such as adolescents, which enables proactive interventions to reduce the risk of hypoglycaemia during exercise. However, it is currently challenging to estimate the duration and intensity of physical activity. Pioneering studies applied wearable devices to measure physiological data, such as heart rate [168] and acceleration [169], as the indicators of physical activity and used them

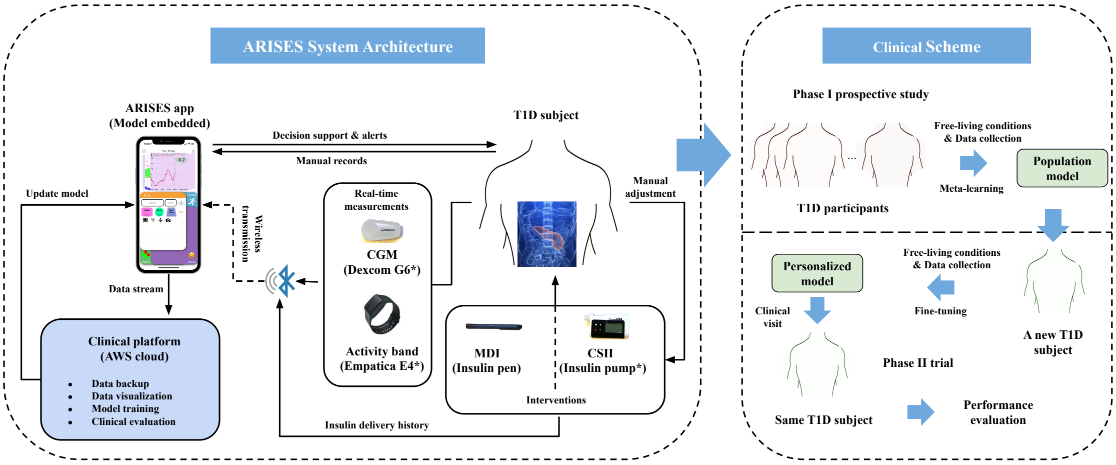


Figure 3.9: System architecture and clinical scheme of ARISES. A T1D subject is equipped with CGM and the wristband to measure glucose levels and vital signs, both of which communicate with the ARISES app via Bluetooth connectivity and provide input data for the deep learning models. The wearable devices in the system are marked by *. The data collected in phase I are used to train a population model with meta-learning, which is then fine-tuned in phase II to develop personalised models.

as exogenous input features of BG predictors. Therefore, in the remaining sections of this chapter, we investigate the effect of physiological data measured by a wearable wristband on BG prediction.

3.5 Non-Invasive Wristband Sensor Data

Wristband sensors have been used in recent literature to estimate physical activity for T1D subjects [170, 171]. In a recent study, wristband measurements combined with food logs were employed as digital biomarkers to estimate interstitial glucose using a machine learning method [172]. However, the clinical efficacy of sensor wristbands in diabetes management remains unproven [173]. The ARISES dataset has been used in Section 3.4 as one of the evaluation datasets to provide CGM data and self-reported events to the FCNN model. In this section, we further analyse the non-invasive wristband sensor data in this proprietary dataset collected in our previous clinical trial. Figure 3.1 shows the overall system architecture and clinical scheme.

3.5.1 Data Collection in Phase I prospective study

This was a six-week longitudinal prospective study (NCT ID: NCT03643692) using a clinically validated real-time physiological data acquisition sensor (Empatica E4) and CGM (Dexcom G6) to identify correlations between measurable physiological parameters and glycaemia. Under free-living conditions, twelve adults (18 years old and older) with a median age (interquartile range (IQR)) of 40 years (30-50) were equally stratified by gender and mode of insulin delivery (MDI and CSII). Participants were recruited from the Imperial College Healthcare NHS trust T1D outpatient clinics, registered research databases, and interested participants who contact us. Throughout the duration of the study, participants wore the Empatica E4 and Dexcom G6 devices with alarm thresholds of glucose levels set at $<$ 4 mmol/L and $>$ 11 mmol/L. Participants were asked to log daily events such as, insulin doses in units, meal macronutrient composition in grams, alcohol intake in units, stress, illness, and exercise in the mySugr smartphone app, which are used to develop the input features of glucose prediction models. All the participants provided written informed consent.

Table 3.7 presents the demographic and clinical characteristics of the 12 T1D participants in the phase I prospective study. We collected a median (IQR) of 1,113.5 (1,059.0-1,184.0) and 832.5 (733.0-953.0) hours of glucose data and sensor wristband data, respectively, and received a total of 5,767 daily entries with a median (IQR) of 396 (237-732.3) interactions (Table B.6 and B.7 of Appendix B), including carbohydrates, protein, fat, insulin bolus, exercise, alcohol, stress, and illness, where the carbohydrate entries account for the largest portion.

3.5.2 Association Between Sensor Data and Adverse Glycaemic Events

Different from most of the previous studies using CGM and daily manual logs [19, 152, 160, 165, 166, 174] and the experiments reported in Section 3.4, the objective of this section is to better understand the effect of the non-invasive physiological data on the prediction of glycaemic events. Using the package lme4 in R, a mixed effects logistic regression [175] was applied calculate the logarithm of odds ratios (ORs) to interpret the relationship

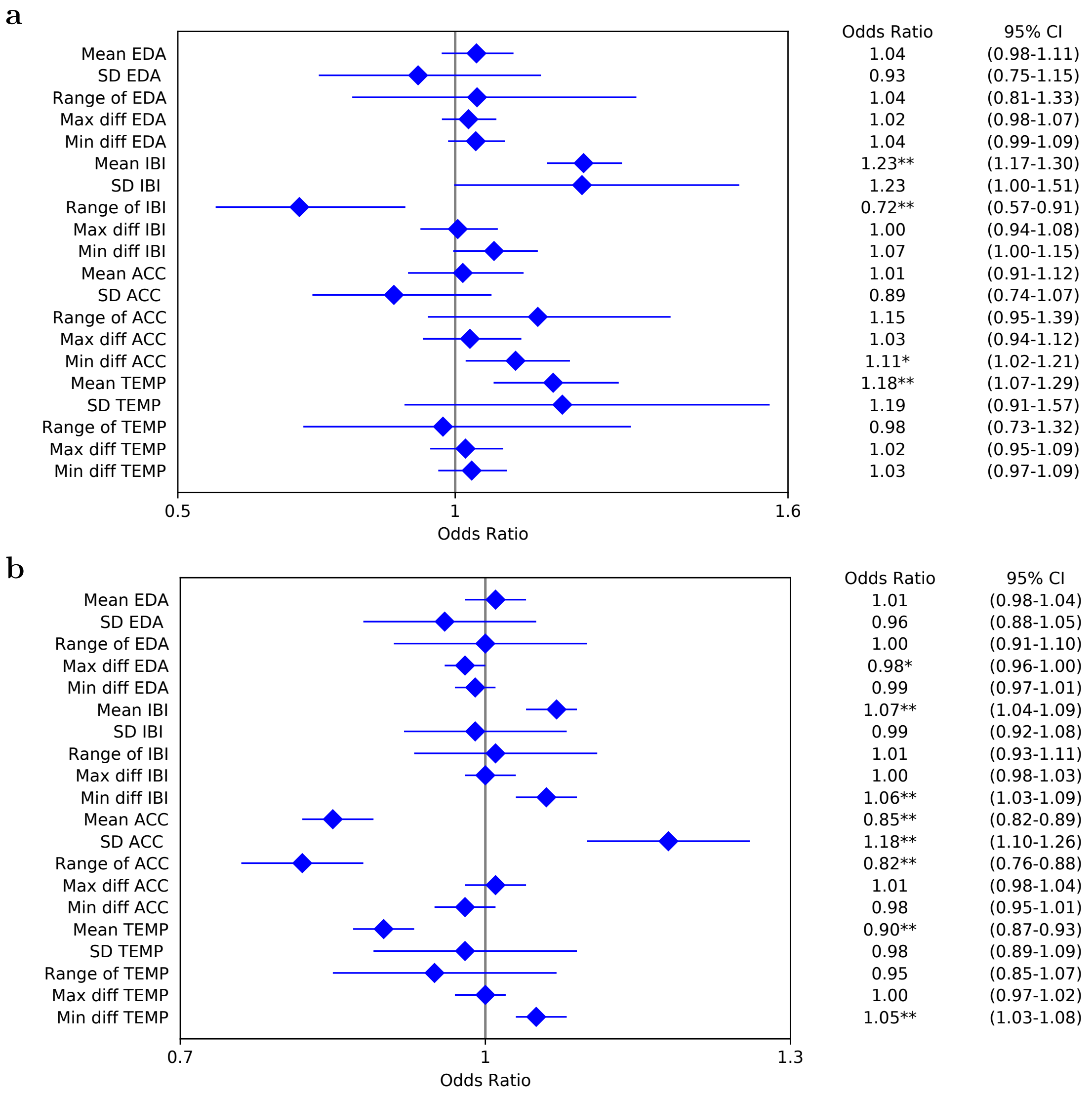
Table 3.7: Demographic characteristics (Median (IQR)) and clinical characteristics (Mean±STD) of the 12 T1D participants in the phase I clinical study

Demographic	
Age (years)	40.0 (30.0-49.0)
Gender (male/female)	6/6 (50.0% male)
Insulin regimen (CSII/MDI)	6/6 (50.0% CSII)
HbA1c (mmol/mol)	50.4 (41.5-57.5)
Glucose data length (hours)	1,113.5 (1,059.0-1,184.0)
Sensor wristband data length (hours)	832.5 (733.0-953.0)
Clinical	
Time below range (< 54 mg/dL) (%)	0.4 ± 0.3
Time below range (< 70 mg/dL) (%)	2.9 ± 1.9
Time in range ([70, 180] mg/dL) (%)	63.4 ± 15.8
Time above range (> 180 mg/dL) (%)	33.7 ± 16.9
Low blood glucose index	0.8 ± 0.5
High blood glucose index	7.6 ± 4.2
Average daily risk range	40.4 ± 10.5
Inter-day coefficient of variation (%)	35.2 ± 4.5
Intra-day coefficient of variation (%)	30.9 ± 4.8
Mean glucose level (mg/dL)	161.2 ± 25.9
Median glucose level (mg/dL)	154.8 ± 26.8

MDI: multiple daily injection, CSII: continuous subcutaneous insulin infusion.

between physiological measurements and the binary outcome of adverse glycaemic events (i.e., hypoglycaemia (< 70 mg/dL) or hyperglycaemia (> 180 mg/dL) in Figure 3.10. The measured physiological variables applied to the regression analysis include the mean values, standard deviation, range, and maximum and minimum differential values of electrodermal activity (EDA), inter-beat intervals (IBIs), acceleration, and skin temperature signals.

The association of the non-invasive physiological measurements with adverse glycaemic events over a 60-minute PH is shown in Figure 3.10. Hypoglycaemia is negatively associated with a larger range of IBIs (OR: 0.72, 95% CI: 0.57-0.91; $p < 0.01$), while higher mean IBIs and mean skin temperature increases the ORs of hypoglycaemia (OR: 1.23, 95% CI: 1.17-1.30; $p < 0.01$; and OR: 1.18, 95% CI: 1.07-1.29; $p < 0.01$, respectively). Similarly, we observe that, besides the IBIs and skin temperature, variables derived from EDA and acceleration are also significant predictive factors for hyperglycaemia prediction. Considering all the physiological signals are significantly associated with adverse glycaemic events, we, therefore, combined these non-invasive measurements with CGM



a

	Odds Ratio	95% CI
Mean EDA	1.04	(0.98-1.11)
SD EDA	0.93	(0.75-1.15)
Range of EDA	1.04	(0.81-1.33)
Max diff EDA	1.02	(0.98-1.07)
Min diff EDA	1.04	(0.99-1.09)
Mean IBI	1.23**	(1.17-1.30)
SD IBI	1.23	(1.00-1.51)
Range of IBI	0.72**	(0.57-0.91)
Max diff IBI	1.00	(0.94-1.08)
Min diff IBI	1.07	(1.00-1.15)
Mean ACC	1.01	(0.91-1.12)
SD ACC	0.89	(0.74-1.07)
Range of ACC	1.15	(0.95-1.39)
Max diff ACC	1.03	(0.94-1.12)
Min diff ACC	1.11*	(1.02-1.21)
Mean TEMP	1.18**	(1.07-1.29)
SD TEMP	1.19	(0.91-1.57)
Range of TEMP	0.98	(0.73-1.32)
Max diff TEMP	1.02	(0.95-1.09)
Min diff TEMP	1.03	(0.97-1.09)

b

	Odds Ratio	95% CI
Mean EDA	1.01	(0.98-1.04)
SD EDA	0.96	(0.88-1.05)
Range of EDA	1.00	(0.91-1.10)
Max diff EDA	0.98*	(0.96-1.00)
Min diff EDA	0.99	(0.97-1.01)
Mean IBI	1.07**	(1.04-1.09)
SD IBI	0.99	(0.92-1.08)
Range of IBI	1.01	(0.93-1.11)
Max diff IBI	1.00	(0.98-1.03)
Min diff IBI	1.06**	(1.03-1.09)
Mean ACC	0.85**	(0.82-0.89)
SD ACC	1.18**	(1.10-1.26)
Range of ACC	0.82**	(0.76-0.88)
Max diff ACC	1.01	(0.98-1.04)
Min diff ACC	0.98	(0.95-1.01)
Mean TEMP	0.90**	(0.87-0.93)
SD TEMP	0.98	(0.89-1.09)
Range of TEMP	0.95	(0.85-1.07)
Max diff TEMP	1.00	(0.97-1.02)
Min diff TEMP	1.05**	(1.03-1.08)

Figure 3.10: Forest plots of mixed effects logistic regression showing the association between non-invasive physiological measurements and adverse glycaemic events, including electrodermal activity (EDA), inter-beat intervals (IBIs), acceleration (ACC), and skin temperature (TEMP). The horizontal error bars represent 95% CIs. The regression coefficients were computed for mean values, standard deviation (SD), range, and maximum and minimum differential (diff) values over a one-hour retrospective window. The differential values refer to difference between adjacent measurements. **a** Analysis for hypoglycaemia. **b** Analysis for hyperglycaemia. The significance of a predictor is indicated as $^{*}p < 0.05$, $^{**}p < 0.01$.

and daily entries to extract a total of 20 real-time features (Table B.8 of Appendix B), which were used in feature selection for the deep learning-based prediction model.

3.6 Integrating Wristband Sensor Data Into BG Prediction

3.6.1 Multi-modal Feature Engineering and Preprocessing

As a clinically validated, commercially available, and non-invasive device, the Empatica E4 wristband uses a photoplethysmography sensor to measure blood volume pulse (BVP), two electrodes to obtain EDA, a pair of accelerometers and a gyroscope to detect the level of physical activity, and a peripheral temperature sensor to monitor skin temperature. In previous clinical studies, BVP and electrocardiogram signals are the primary sources to identify IBIs for heart rate variability analysis [176]. In particular, we applied a band-pass Butterworth filter to remove noise in BVP signals and employed a slope sum function [177] to detect the local maxima. Then we used a sliding window with decision rules [177] to search peaks, as the systoles in cardiac cycles. The IBIs were computed by the difference of consecutive peaks.

We extracted short-term heart rate variability features with a 5-minute window to indicate early changes [178] in temporal and frequency domains [179]. To obtain skin conductance levels and skin conductance responses, we continuously decomposed EDA data into tonic and phasic components via a high-pass filter [180]. There are two open-source software tools involved in EDA and BVP processing [181, 182]. Together with physical activity levels and skin temperature, the outcomes of these features in the past five minutes were averaged and aligned with the time steps of CGM readings. Heart rate variability is an established indicator that reflects cardiac autonomic activities, while EDA is related to the status of the nervous system. These biomarkers have been used in previous studies to predict and detect hypoglycaemia for T1D [178, 183, 184].

The median value of the missing percentages of CGM and wristband data are 3.02% and 23.05%, respectively, which are reasonable since the wristband needs to be charged for around 4-5 hours every day. All the features, including IOB and COB, were preprocessed following the procedure in Section 3.2.1. We noted that features derived from the same measurement exhibited strong a correlation with each other. Hence, each time we retained one feature in IBI or EDA feature group (Table B.8 of Appendix B) and selected the best combination according to the error scores that summed up RMSE results for the four PHs in model validation.

Considering the personalised models are provided to the T1D subjects at a midterm clinical visit (Figure 3.9), we divided the data of each subject into a training set and a testing set that include the first 50% data and the remaining 50% data, respectively. The last 20% data of each training set were used as a hold-out personalised validation set. To simulate a clinical scheme with two phases (Figure 3.9), we employed a population set containing the training sets of 11 subjects and a personalised set with the data of the remaining subject, assuming it is a new subject (e.g., a participant in phase II). Data of each subject in the population set were used to optimise the population model. The population models were validated with leave-one-subject-out cross validation. Then we used the training data of the personalised set to fine-tune the population model to develop a personalised model, and used the testing data of the personalised set for evaluation. We used early stopping to mitigate overfitting and improve generalization.

The chronological partition of training, validation and testing set in this work was carefully selected. Random cross-validation can be found in previous work, which trained and validated machine learning models on the same dataset [131, 152]. However, during the experiments, we noticed that there were temporal dependencies between the data points from nearby locations, especially in adjacent ones. The features were derived with the small resolution of CGM, so the difference between consecutive time steps is sometimes negligible. In this regard, the use of random or stratified splitting methods would introduce underlying temporal correlation into training and testing sets, which could result in serious overestimation of model accuracy [185].

3.6.2 Developing Population and Personalised models

With the population and personalised data sets, we applied the FCNN framework (Section 3.2) to develop a deep learning model for glucose prediction and hypo- and hyperglycaemia detection. As shown in Figure 3.11, the architecture of the ARISES model is similar to the attention-based RNN with GRUs proposed in Section 3.2.2, where a single

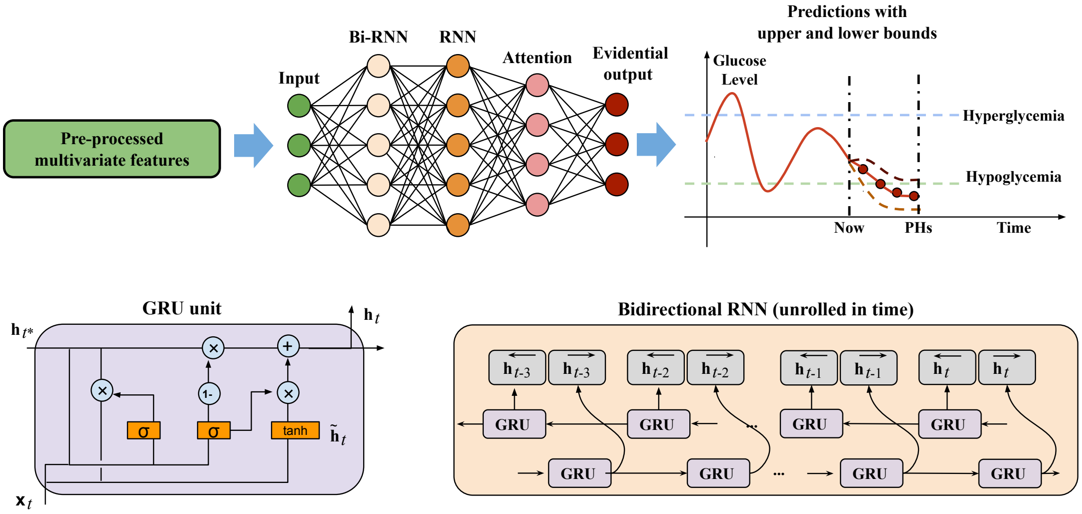

Pre-processed multivariate features
Input
Bi-RNN
RNN
Attention
Evidential output
Predictions with upper and lower bounds
Glucose Level
Hyperglycemia
Hypoglycemia
Now
PHs
Time
GRU unit
$\mathbf{h}_{t^*}$
$\mathbf{h}_t$
$\tilde{\mathbf{h}}_t$
$\mathbf{x}_t$
1-
σ
σ
tanh
Bidirectional RNN (unrolled in time)
$\overleftarrow{\mathbf{h}}_{t-3}$
$\overrightarrow{\mathbf{h}}_{t-3}$
$\overleftarrow{\mathbf{h}}_{t-2}$
$\overrightarrow{\mathbf{h}}_{t-2}$
$\overleftarrow{\mathbf{h}}_{t-1}$
$\overrightarrow{\mathbf{h}}_{t-1}$
$\overleftarrow{\mathbf{h}}_{t}$
$\overrightarrow{\mathbf{h}}_{t}$
GRU
...

Figure 3.11: Architecture of the ARISES model to incorporate wristband sensor data.

GRU layer was used to replace the second and third bidirectional GRU layers in Figure 3.2. The multivariate input data for the RNN model were selected according to validation performance, which include CGM, carbohydrate amount, insulin bolus, time index, IBIs, and skin conductance responses. Due to the change of input data and experimental settings, the hyperparameters were updated and presented in Table B.3 of Appendix B. Interestingly, according to Figure 3.10, the IBI also has significant effects on both hypoglycaemia and hyperglycaemia prediction in mixed effects logistic regression.

In addition, lower bounds B^l_{t+r} and upper bounds B^u_{t+r} were applied to improve hypo- and hyperglycaemia prediction, respectively, which are denoted as

$$\begin{aligned} B^l_{t+r} &= \hat{G}_{t+r} - k^l f_N^{-1}(u^e_t), \\ B^u_{t+r} &= \hat{G}_{t+r} + k^u f_N^{-1}(u^e_t), \end{aligned} \tag{3.29}$$

where k^l and k^u are tunable thresholds of the uncertainty.

3.6.3 Prediction Performance

We used the results for the 60-minute PH as the primary outcomes, since predicting glucose over such a long PH is challenging. The converted TensorFlow Lite models were evaluated to simulate on-device inference in the ARISES app (Section 6.5.3). To compare the proposed model with existing approaches, we employed a set of classic machine learning and deep learning baseline methods, including SVR with the radial basis function kernel [160], ANNs with three fully-connected layers [174], Bi-LSTM [89], and CRNNs [96]. Besides, we also used a statistical model, the autoregressive moving average (ARMA) with exogenous inputs [186], and a physiological model, the physiologically-based kinetic model (PKM), which is based on the composite minimal model of plasma glucose and insulin kinetics with personalised insulin sensitivity, time to maximum glucose rate of appearance, and time to maximum insulin absorption [187, 188]. The PKM has been validated on both the *in silico* data from the UVA/Padova T1D simulator [81] and real data from clinical trials [189] in terms of glucose prediction [188]. The input features of baseline models were identical to those of the proposed model, except that the PKM only used the information of CGM measurements, carbohydrate intake and insulin bolus.

Figure 3.12 depicts the predicted trajectories and CGM measurements of a participant over a two-day period. We present the 7-day trajectories of four selected participants in Figure B.2 of Appendix B. We observe that the daily activities, including meal intake and insulin bolus delivery, have a significant impact on the glucose levels. The glycaemic homeostasis is affected by these external factors and internal changes in the T1D subject. Thus, the accuracy degrades as the PH increases. As highlighted by the eclipses in Figure 3.12, the use of lower bounds successfully identified two hypoglycaemic events that are likely to be missed using single prediction values.

Glucose Level Prediction

Table 3.8 presents the results (Mean±STD) of the personalised models with 15, 30, 45, 60-minute PHs. The proposed ARISES model outperformed all the considered baseline methods in terms of RMSE, gRMSE, MAE, MAPE, and PTD. The results of the baseline

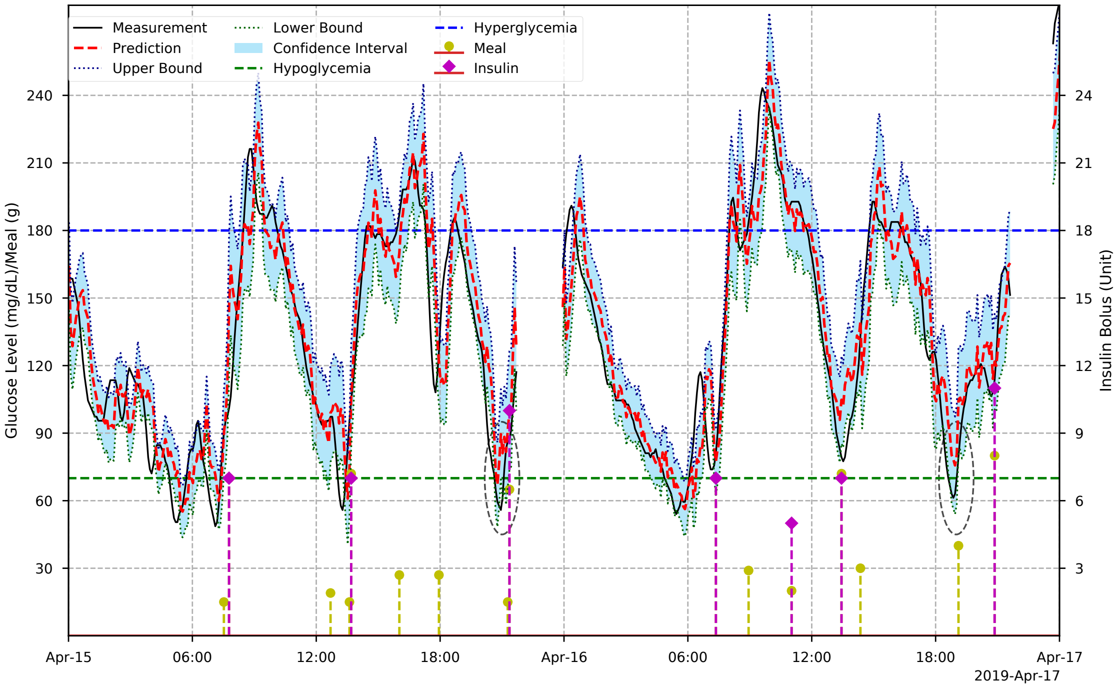


Figure 3.12: Two-day period CGM and prediction trajectories of a T1D adult over a 30-minute PH. The ellipses indicate the hypoglycaemic events that are missed by prediction values but detected by the lower bounds.

methods are presented in Table 3.9. The ARISES obtained significant improvement in RMSE, gRMSE, MAE, and MAPE, when compared with the best performance of the baseline methods (CRNNs [96]; $p < 0.01$). When only one day of data was used for fine-tuning, the MAML approach obtained the average RMSE of 39.37 ± 7.14 for the 60-minute PH, which is much smaller than the RMSE obtained by a baseline method of transfer learning [128] (RMSE: 43.07 ± 8.41; $p < 0.05$).

Table 3.8: Performance of glucose level prediction evaluated on 12 clinical subjects

PHs	15 min	30 min	45 min	60 min
RMSE (mg/dL)	10.15 ± 1.67	20.92 ± 3.55	28.99 ± 4.41	35.28 ± 5.77
gRMSE (mg/dL)	12.14 ± 2.06	26.07 ± 4.47	37.20 ± 5.97	46.26 ± 7.73
MAE (mg/dL)	7.21 ± 1.09	15.06 ± 2.36	21.15 ± 3.15	26.11 ± 4.36
MAPE (%)	5.07 ± 0.97	10.62 ± 2.03	14.94 ± 2.77	18.53 ± 3.78
PTD (min)	1.39 ± 1.06	7.37 ± 5.18	14.00 ± 7.24	17.63 ± 11.39

In addition, Figure 3.13 shows the results of ablation analysis, where we removed certain components from the model and evaluated their impact on the prediction perfor-

Table 3.9: Performance of baseline methods for glucose level prediction over the 60-minute PH

Methods	RMSE ↓	gRMSE ↓	MAE ↓	MAPE ↓	PTD ↓
ARISES	35.28 ± 5.77	46.26 ± 7.73	26.11 ± 4.36	18.53 ± 3.78	17.63 ± 11.39
PKM [188]	$43.12 \pm 7.53^\dagger$	$51.22 \pm 10.42^\dagger$	$31.51 \pm 5.09^\dagger$	$21.95 \pm 4.28^\dagger$	$19.87 \pm 10.26^*$
ARMA [186]	$42.29 \pm 7.73^\dagger$	$53.44 \pm 10.24^\dagger$	$30.77 \pm 5.43^\dagger$	$21.04 \pm 3.96^\dagger$	$41.89 \pm 13.19^\dagger$
SVR [160]	$39.99 \pm 7.03^\dagger$	$52.03 \pm 10.48^\dagger$	$31.52 \pm 5.93^\dagger$	$24.19 \pm 5.61^\dagger$	21.34 ± 11.49
ANN [174]	$39.31 \pm 6.85^\dagger$	$51.88 \pm 9.59^\dagger$	$29.69 \pm 5.35^\dagger$	$21.15 \pm 4.10^\dagger$	$31.88 \pm 13.65^\dagger$
Bi-LSTM [89]	$39.32 \pm 7.19^\dagger$	$51.62 \pm 9.92^\dagger$	$29.02 \pm 5.23^\dagger$	$20.21 \pm 4.01^\dagger$	19.57 ± 10.71
CRNN [96]	$37.18 \pm 6.09^\dagger$	$49.04 \pm 8.50^\dagger$	$27.77 \pm 4.89^\dagger$	$19.55 \pm 3.70^\dagger$	21.57 ± 11.41

$^*p < 0.05$, $^\dagger p < 0.01$.

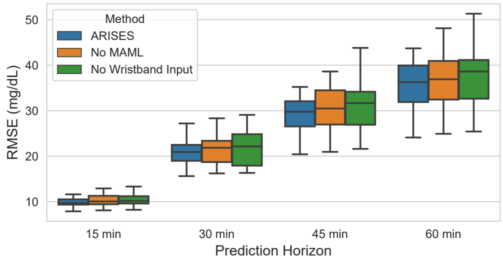


Figure 3.13: Ablation analysis on the prediction performance of glucose levels. The model achieved smaller average RMSE for the 12 T1D subjects when using MAML and wristband input data. The improvement is most significant for the 60-minute PH. The lower and upper hinges of boxplots show the first quarter (Q1) and the third quartile (Q3), respectively. The central lines indicate the median, while the whiskers extend to 1.5 IQR.

mance. In particular, the use of MAML and wristband input respectively reduced the average RMSE by 1.41 and 2.25 mg/dL ($p < 0.05$) for the 60-minute PH.

Hypoglycaemia and Hyperglycaemia Prediction

Table 3.10 and 3.12 respectively show the results of hypoglycaemia and hyperglycaemia prediction using the lower and upper bounds derived from evidential deep learning. We observe that the proposed ARISES model achieved the accuracy of 88.58% with the sensitivity of 70.30% and the accuracy of 87.20% with the sensitivity of 86.62% for hypoglycaemia and hyperglycaemia prediction over the 60-minute PH, respectively. For the considered baseline methods (Table 3.11 and 3.13), we used the predicted glucose levels, i.e., single trajectories, to detect adverse glycaemic events. Among these, ARMA [186] and PKM [188] achieved the best baseline results for hypoglycaemia and hyperglycaemia prediction, respectively. It is worth noting that, compared with the ARMA, the ARISES model significantly increased sensitivity by 13.35% and reduced MD by 13.29 mg/dL for hypoglycaemia prediction. Compared with the PKM for hyperglycaemia prediction, the ARISES model significantly increased specificity and precision by 5.38 % and 5.43 %, respectively, while reducing the MD by 13.80 mg/dL. As shown in Figure 3.14, we observe that the use of lower bounds and wristband input data enhanced the average MCC scores by 0.34 ($p < 0.01$) for hypoglycaemia prediction with the 60-minute PH.

Table 3.10: Performance of hypoglycaemia prediction evaluated on 12 clinical subjects

PHs	15 min	30 min	45 min	60 min
Accuracy (%)	98.03 ± 1.03	94.96 ± 2.92	91.97 ± 4.22	88.58 ± 6.53
Sensitivity (%)	84.15 ± 4.20	76.08 ± 5.88	72.07 ± 4.45	70.30 ± 12.84
Specificity (%)	98.72 ± 0.75	96.42 ± 2.48	93.99 ± 4.08	90.09 ± 8.21
Precision (%)	78.91 ± 4.31	65.65 ± 5.31	58.23 ± 10.21	56.20 ± 10.43
MCC score	0.80 ± 0.04	0.68 ± 0.05	0.60 ± 0.06	0.56 ± 0.07
MD (mg/dL)	10.28 ± 3.80	19.18 ± 7.00	26.30 ± 9.28	28.63 ± 11.00

Table 3.11: Performance of hypoglycaemia prediction with the baseline methods over the 60-minute PH

Methods	Accuracy ↑	Sensitivity ↑	Specificity ↑	Precision ↑	MCC score ↑	MD ↓
ARISES	88.58 ± 6.53	70.30 ± 12.84	90.09 ± 8.21	56.20 ± 10.43	0.56 ± 0.07	28.63 ± 11.00
PKM [188]	$80.05 \pm 6.29^{\dagger}$	$82.49 \pm 13.10^{*}$	$78.75 \pm 8.77^{\dagger}$	$36.43 \pm 12.98^{\dagger}$	$0.45 \pm 0.11^{*}$	$40.36 \pm 15.34^{\dagger}$
ARMA [186]	91.89 ± 5.23	$56.95 \pm 19.24^{\dagger}$	94.87 ± 3.67	55.28 ± 17.38	0.51 ± 0.12	$41.92 \pm 14.60^{\dagger}$
SVR [160]	87.20 ± 6.08	$6.03 \pm 10.16^{\dagger}$	$99.10 \pm 1.40^{\dagger}$	30.05 ± 32.98	$0.08 \pm 0.12^{\dagger}$	$60.55 \pm 22.73^{\dagger}$
ANN [174]	88.01 ± 5.80	$17.15 \pm 11.75^{\dagger}$	$98.39 \pm 1.43^{\dagger}$	61.76 ± 15.16	$0.26 \pm 0.11^{\dagger}$	$48.18 \pm 18.05^{\dagger}$
Bi-LSTM [89]	87.51 ± 6.65	$19.78 \pm 12.38^{\dagger}$	$97.52 \pm 1.92^{*}$	47.79 ± 20.69	$0.25 \pm 0.15^{\dagger}$	$43.61 \pm 16.73^{\dagger}$
CRNN [96]	88.06 ± 5.94	$12.54 \pm 8.40^{\dagger}$	$99.07 \pm 0.72^{\dagger}$	67.70 ± 16.88	$0.24 \pm 0.11^{\dagger}$	$45.94 \pm 19.37^{\dagger}$

$^{*}p < 0.05$, $^{\dagger}p < 0.01$.

Table 3.12: Performance of hyperglycaemia prediction evaluated on 12 clinical subjects

PHs	15 min	30 min	45 min	60 min
Accuracy (%)	96.75 ± 0.99	93.22 ± 1.24	90.06 ± 1.05	87.20 ± 1.95
Sensitivity (%)	95.32 ± 2.16	91.25 ± 4.75	88.48 ± 7.87	86.62 ± 7.81
Specificity (%)	96.95 ± 1.14	92.62 ± 2.89	87.61 ± 5.10	82.59 ± 7.96
Precision (%)	94.62 ± 2.37	90.51 ± 3.20	87.43 ± 4.76	85.11 ± 5.83
MCC score	0.92 ± 0.02	0.84 ± 0.02	0.77 ± 0.03	0.70 ± 0.05
MD (mg/dL)	13.00 ± 2.30	28.69 ± 5.16	40.05 ± 8.04	47.62 ± 10.33

Table 3.13: Performance of hyperglycaemia prediction with the baseline methods over the 60-minute PH

Methods	Accuracy ↑	Sensitivity ↑	Specificity ↑	Precision ↑	MCC score ↑	MD ↓
ARISES	87.20 ± 1.95	86.62 ± 7.81	82.59 ± 7.96	85.11 ± 5.83	0.70 ± 0.05	47.62 ± 10.33
PKM [188]	85.54 ± 3.13	91.58 ± 3.52	77.21 ± 4.74*	79.68±12.49*	0.68 ± 0.06	61.42±16.26†
ARMA [186]	85.05 ± 1.85†	82.26 ± 7.02†	83.52 ± 6.27	84.59 ± 6.51	0.66 ± 0.04†	65.36±14.47†
SVR [160]	84.73 ± 3.24†	83.44 ± 15.68	75.59 ± 19.79	84.04 ± 5.22	0.64 ± 0.08†	52.09±10.85*
ANN [174]	84.13 ± 2.15†	76.24±18.61*	82.43 ± 13.46	85.77 ± 4.85	0.62 ± 0.08†	56.65±11.12†
Bi-LSTM [89]	84.22 ± 3.38†	77.03±17.55*	83.19 ± 11.59	85.84 ± 4.58	0.63 ± 0.08†	57.43±10.60†
CRNN [96]	83.97 ± 4.33†	76.03±20.07*	82.70 ± 14.53	87.28 ± 4.57	0.63 ± 0.09†	53.47±10.27*

$^{*}p < 0.05$, $^{\dagger}p < 0.01$.

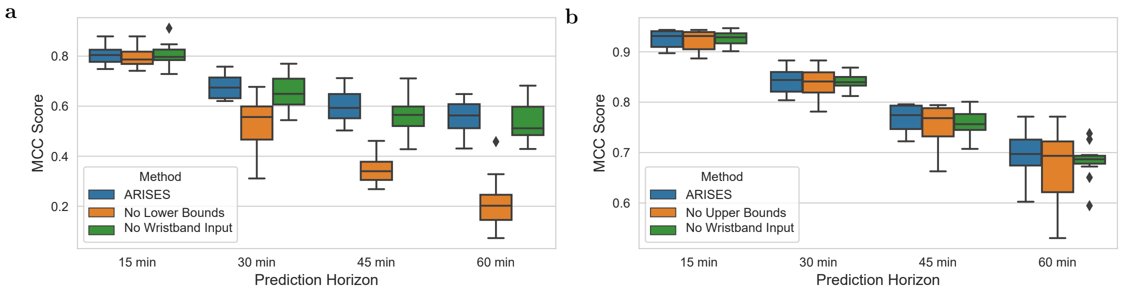


Figure 3.14: Ablation analysis on the prediction of adverse glycaemic events, evaluated on the 12 T1D subjects. **a** MCC scores for hypoglycaemia prediction. **b** MCC scores for hyperglycaemia prediction. The lower bounds significantly improved hypoglycaemia prediction, while the use of wristband data enhanced MCC scores for all the PHs. The lower and upper hinges of boxplots show the Q1 and the Q3, respectively. The central lines indicate the median, while the whiskers extend to 1.5 IQR.

It is observed that the machine learning and deep learning baseline models obtained better RMSE performance for glucose level prediction, but smaller MCC scores for hypo- and hyperglycaemia prediction, when compared with the physiological and statistical baseline methods. One possible explanation is that the machine learning and deep learning baseline models were optimised in a supervised learning process with the targets of actual CGM measurements, but the prediction of adverse glycaemic events was not considered. In this regard, the introduced lower and upper bounds in the ARISES model enabled a good balance between glucose level prediction and hypo- and hyperglycaemia detection.

We compared the MCC scores using these bounds against the results of single curve prediction in Figure 3.14, where the classification based on the bounds exhibited better performance. We noticed that hypoglycaemia is a minority class in the dataset, which accounts for 2.91 ± 1.93% of total glucose measurements (Table 3.7). In general, the classifier is less sensitive to detecting a minority class. Nevertheless, in this model, the sensitivity can be further enhanced by reducing the thresholds of lower bounds at a cost of potential alarm fatigue. This trade-off can be decided by clinicians on an individual case basis.

3.7 Conclusion

In this chapter, we proposed FCNN, a novel deep learning framework to overcome the critical clinical challenges in personalised BG prediction: confidence in the predictions and data availability. Incorporating meta-learning and evidential deep learning, we developed a fast-adaptive and confident deep learning model based on the bidirectional GRU and attention mechanism. To the best of our knowledge, FCNN is the first work that uses an attention-based GRU model for BG prediction, while incorporating model confidence and fast adaptation for clinical benefits. We have shown that FCNN achieved state-of-the-art performance and significantly outperformed the selected baseline methods, in terms of RMSE, MAE, and gRMSE, when evaluated on the three clinical datasets. The confidence bounds derived by evidential regression have notably improved MCC scores of hypoglycaemia detection, especially for the 60-minute PH. Compared with the classic TL baseline, the use of MAML has obtained superior performance of fast adaptation when training data are limited.

We further integrated wristband sensor data into the FCNN to build a deep learning predictor for the ARISES clinical study. The results suggest that measurements obtained from wearable wristband data sensors can be integrated alongside CGM data to improve the prediction of glucose levels and adverse glycaemic events. It is also noted that indicate heart rate variability can be useful biomarkers in T1D decision support.

Chapter 4

Glycaemic Control Using Deep Reinforcement Learning

People with T1D require the regular exogenous infusion of insulin to maintain their BG concentration in a therapeutically adequate target range. Although the AP and CGM have been proven to be effective in achieving closed-loop control, significant challenges still remain due to the high complexity of glucose dynamics and limitations in the technology. Optimising the doses of insulin delivery to minimise the risk of hyperglycaemia and hypoglycaemia is still an open problem. In this context, this chapter first presents a novel DRL model for single-hormone (basal insulin) and dual-hormone (basal insulin and glucagon) delivery. In particular, the delivery strategies are developed by double Q-learning. For designing and testing purposes, the FDA-accepted UVA/Padova T1D simulator is employed. *In silico* results show that the single and dual-hormone delivery strategies achieve good glycaemic control when compared with low-glucose insulin suspension. The simulator is further used to develop a novel DRL advisor for meal insulin bolus using an actor-critic model and prioritised memory replay. The DRL insulin bolus advisor significantly improved the time spent in the adverse glycaemic events for virtual cohorts of adults and adolescents.

DRL has provided new paradigms of glycaemic control algorithms. However, all the existing DRL-based AP controllers require a large number of random online interactions between the agent and environment, which can be validated in T1D simulators but are not feasible in clinical settings. To this end, we propose an offline DRL framework that can develop and validate models purely offline. We evaluated the proposed framework on an offline *in silico* dataset and a clinical dataset with 12 real T1D subjects. The performance on the *in silico* dataset shows that the offline DRL algorithm significantly increased TIR for the adult and adolescent groups. A notable increase in policy values was observed for each subject in the clinical dataset. The work in this chapter has led to the following journal articles:

- **T. Zhu**, K. Li, P. Herrero and P. Georgiou, ”Basal glucose control in type 1 diabetes using deep reinforcement learning: An in silico validation,” *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 4, pp.1223–1232, 2021.
- **T. Zhu**, K. Li, L. Kuang, P. Herrero and P. Georgiou, “An insulin bolus advisor for type 1 diabetes using deep reinforcement learning,” *Sensors*, vol. 20, no. 18, p. 5058, 2020. (**Editor’s Choice Article**)
- **T. Zhu**, K. Li, P. Herrero and P. Georgiou, ”Offline Deep Reinforcement Learning and Off-Policy Evaluation for Personalized Blood Glucose Control in Type 1 Diabetes,” Submitted to *IEEE Journal of Biomedical and Health Informatics.*

4.1 Introduction

To date, most existing AP systems that have been evaluated in the clinic have used a control engineering approach for basal insulin control [190], such as the model predictive control [191, 192], and proportional integral control [193]. Other groups have also employed a bio-inspired approach [56] and a conventional AI approach (fuzzy logic) [194]. In particular, two of them, the Medtronic MiniMed 670G and the Tandem Control-IQ have reached the commercialization stage. More recently, CamAPS FX and MiniMed 780G obtained Conformité Européenne marking and were commercialized in 2020. Two dual-hormone AP systems, Beta Bionics iLet and Inreda APS, have been shown to improve glycaemic control in clinical trials [195, 196] and are in preparation for commercialization.

For meal bolus delivery, different research groups have developed advanced bolus advisors to further enhance the accuracy of insulin dose recommendations. Assuming the bolus insulin therapy in people with T1D is repetitive by nature, the use of run-to-run control with capillary blood measurements has been proposed [197] and clinically evaluated [198]. By enhancing run-to-run control with CGM measurements and the AI technique of case-based reasoning, Herrero *et al.* proposed an advanced bolus advisor which was integrated into a smartphone [199] and clinically evaluated in a free-living setting over six weeks [200]. In recent years, taking advantage of the increasing data availability thanks to the use of wearables and EHRs, AI is playing an important role in decision support systems for diabetes management [201]. Tyler *et al.* proposed a KNN decision support system to provide weekly insulin dosage recommendations to T1D adults with MDI therapy [202]. The use of the KNN classifier was compared with rule-based reasoning for meal, insulin, and exercise recommendation in [203]. In [204], Aiello *et al.* proposed a KNN classification algorithm to predict postprandial glucose profiles and suggest corrections to meal insulin. DNNs have been employed to determine the insulin bolus dose [205], and a reinforcement learning method was employed to personalise insulin treatment [206]. Combined with a reinforcement learning control algorithm, the GoCARB system was presented within an AP framework to estimate carbohydrates of meals to improve glycaemic control [207]. However, although these systems have been proven to improve glycaemic control [193, 208], challenges remain, and further work is needed to achieve optimal therapeutic targets.

In the recent systematic review by Tejedor *et al.* on the application of reinforcement learning to BG control [209], almost all the included studies (i.e., 29 out of 30) employed traditional reinforcement learning approaches, except for a recent work using a DRL algorithm [102], which is compared with our work in Section 4.2.4. Unlike successful DRL applications in the virtual world, such as Atari video games [80] or board-game Go [76], where an agent dynamically interacts with a virtual environment, performing such exploration on human subjects can be dangerous without proper safety supervision. Alternatively, DRL algorithms can learn from existing collected data using experience replay. This process is called off-policy learning and plays an important role in practical scenarios. However, collecting the training data required is expensive and time-consuming [210]. Fortunately, the UVA/Padova T1D simulators, including S2008 [103] and S2013 [81] versions, provide a perfect platform that allows the agents to freely explore different actions or exploit learned policy. The S2013 version is upgraded from the S2008 version by incorporating a nonlinear hypoglycaemia model and counter-regulation (i.e., glucagon administration). These simulators were accepted by FDA for pre-clinical *in silico* trials and have been proved to match the observations in actual clinical trials [211]. However, the simulators have several limitations [81]. First, many intra-subject variability and realistic scenarios, such as physical activity, complex carbohydrates, and recurrent illness, are not included in current versions. Secondly, physiological models, such as glucagon kinetics, and hardware models, such as CGM and insulin pumps, need to be further updated to better fit the data collected in recent clinical trials.

Thus, most existing studies on glucose control and DRL developed algorithms in the UVA/Padova T1D simulator and reported primary outcomes based on *in silico* population. Employing the UVA/Padova T1D simulator (S2008), DQNs [102] and proximal policy optimization [212] were proposed to control basal-bolus insulin with discrete insulin doses. Similarly, Fox *et al.* [213] proposed a soft actor-critic algorithm with continuous action space, which exhibited low glycaemic risk on 2.1 million hours of simulated data. In [214], the authors used Hovorka model [215] for *in silico* simulation and developed a trust-region policy optimization algorithm for basal insulin control. The work in this chapter is mainly based on the UVA/Padova T1D simulator (S2013) with 10 virtual adults and 10 virtual adolescents. In our previous work on glycaemic variability [216], we introduced modifications to the simulator to emulate realistic intra-subject variability and uncertainty for the adult and adolescent subjects. In this research, we also apply these modifications to the simulator, and the children cohort is currently not included.

Several notable limitations hamper the widespread adoption of DRL in actual T1D management. First, all these existing approaches were developed by online learning in T1D simulators and required long-term error and trial exploration, which is possible in virtual simulation but impractical and dangerous in clinical settings. Secondly, none of these studies evaluated the algorithms on real clinical datasets and failed to prove the generalization of the DRL models, due to the lack of methods for off-policy evaluation (OPE). Fortunately, the recent advances in offline DRL [217, 218, 219, 220, 221] and OPE [222, 223, 224] respectively address the problem of policy learning and policy evaluation on fixed historical datasets instead of online interactions with environments. A major challenge for offline DRL is the trade-off between distributional shift and policy improvement, which is generally tackled by either using policy constraints [217, 218, 219] or value function regularization [220, 221]. Fitted Q evaluation (FQE) [223] is a promising OPE method that has been demonstrated to provide accurate policy value estimates for several large benchmark datasets [225, 226], as well as for a healthcare application of sepsis treatment [227].

This chapter consists of the following three parts: single-hormone and dual-hormone control in Section 4.2, meal insulin bolus recommendation in Section 4.3, and basal insulin control in Section 4.4. In particular, the DRL algorithms are developed and evaluated in the UVA/Padova T1D simulator in Section 4.2 and 4.3, while a clinical dataset, the OhioT1DM dataset, is used in Section 4.4 to validate the proposed offline DRL algorithm.

4.2 Single and Dual-Hormone Basal Glucose Control

Figure 4.1 depicts an overview of the system architecture used to develop the DQN controllers evaluated on the T1D *in silico* environment and to be potentially used in clinical trials. Algorithm 1 and Algorithm 2 correspond to the two-step learning framework described in Section 4.2.2.

4.2.1 Problem Formulation

The problem of basal glucose closed-loop control in T1D can be formulated as a Markov decision process with noise, which is defined by a tuple $\langle S, \mathcal{P}, A, R, \gamma \rangle$ consisting of a state S (i.e., physiological state), a state transition function $\mathcal{P}$ (i.e., physiological model), an action A (i.e., insulin and glucagon control actions), a reward function R (i.e., glycaemic

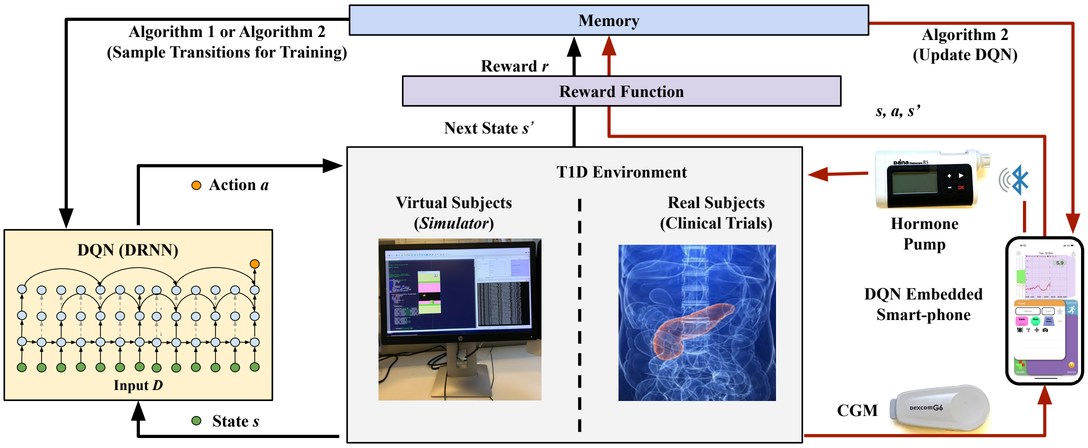


Figure 4.1: System architecture to implement DQN for basal glucose control in the T1D simulator (black arrows) and clinical trials (red arrows).

outcomes), and a discount factor $\gamma \in [0, 1]$ (i.e., the importance of future glycaemic outcomes). The agent in the environment takes an action $a \in A$ at each time step (i.e., each CGM measurement), and then its state $s \in S$ turns into the successor state s' according to $\mathcal{P}$. The policy to select action for given states is denoted by π. Maximising the accumulation of expected reward is the target of reinforcement learning. An action-value (Q-function) $Q^\pi(s, a)$ can be defined to compute this reward:

$$Q^\pi(s, a) = \mathbb{E}[\sum_{t'=t+1}^{\infty} \gamma^{t'-t-1} r_{t'} | s_t = s, a_t = a, \pi]. \tag{4.1}$$

The optimal action-value function $Q^*(s, a) = \max_\pi Q^\pi(s, a)$ offers the maximal values, which can be determined by solving the Bellman equation defined by

$$Q^*(s, a) = \mathbb{E}_{s'}\left[R(s, a) + \gamma \max_{a'} Q^*(s', a')\right], \tag{4.2}$$

The optimal action-value at the current state s is obtained by selecting the action that maximises expected return with the optimal $Q^*(s', a')$ at the next state s'. Although this recursive equation can be estimated by an iterative update, linear and non-linear approximators are commonly used in reinforcement learning for better generalization [80].

In this paper, DQNs are employed to approximate the action-values $Q(s, a; \theta) \approx Q^*(s, a)$ where θ represents the parameters of the neural networks.

Agent states

In the closed-loop glucose control problem, we collect the multi-modal data from the control system, as shown in Figure 4.1, to form a multi-dimensional input vector D to approximate physiological state S. Specifically, D comprises the real-time continuous blood glucose levels G (mg/dL) measured with a CGM sensor, the carbohydrate estimation of meal ingestion M (g) recorded through a smartphone app, and hormone doses delivered by the infusion pumps, including the meal bolus insulin B, basal insulin Bas, and glucagon dose C. Thus, we have $D = \{G, M, I, C\} = [d_{t+1-L}, \cdots, d_t]^{\mathbb{T}} \in \mathbb{R}^{\mathbb{L} \times 4}$, where L is the length of the time steps vector, $I = B + Bas$ (Unit) represents the sum of meal bolus insulin and basal insulin. The approximated observation $o_t = s_t + e_t$ takes into account the errors or miss-estimations e_t in glucose measurements G, carbohydrate meal estimation M, and the meal insulin bolus B. Here B is computed from M with a standard bolus calculator (SBC) [228]. From a DRL perspective, the problem can be seen as an agent interacting with an environment over sequential time steps. Every five minutes, an observation o_t can be obtained from the environment, and an action a_t can be taken according to the agent's policy. We choose a five-minute time scale because this is the common sampling frequency for many commercial CGMs (e.g. Dexcom G6; Medtronic Guardian) and a typical setting for AP systems [229]. Glucose-insulin-glucagon dynamics are quite slow; hence a shorter sampling period is unlikely to improve the outcomes of an AP system.

Actions

Following the same framework, we provide two types of delivery strategies for different pump settings. For people with T1D wearing insulin pumps, the action space is defined by modifying the basal insulin rate (BR) as follows: {suspension of BR, 0.5*BR, BR, 1,5*BR, 2*BR}. For those wearing dual-hormone pumps, the action space is defined by the following options: {suspension of BR, 0.5*BR, BR, 1,5*BR, 2*BR, delivering glucagon}.

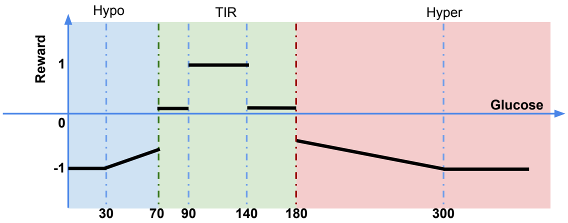


Figure 4.2: Visualization of the employed reward function in terms of the glucose level (mg/dL) in the next state.

The value of BR is subject-specific and is known in advance. Based on previous work, we fix glucagon doses to 0.3 µg/kg for all individuals and constraint the total amount of delivered glucagon to a maximum of one mg per day [230]. This dosage has also been tested in clinical trials with two formulations of glucagon, which demonstrates efficacy and safety [231].

Rewards

The desired performance of closed-loop glucose control is to maintain BG in a target range of 70-180 mg/dL. By using an empirical approached aiming at maximising TIR and minimising hypoglycaemia, the following piece-wise reward function was selected.

$$r_{t+1} = \begin{cases} 1, & 90 \leq G_{t+1} \leq 140 \\ 0.1, & 70 \leq G_{t+1} < 90 \ \& \ 140 < G_{t+1} \leq 180 \\ -0.4 - (G_{t+1} - 180)/200, & 180 < G_{t+1} \leq 300 \\ -0.6 + (G_{t+1} - 70)/100, & 30 \leq G_{t+1} < 70 \\ -1, & \text{else.} \end{cases} \tag{4.3}$$

As depicted in Figure 4.2, the agent receives a positive reward if the BG level for the next state is in the target range and a negative reward otherwise. If the BG is below 30 mg/dL or above 300 mg/dL, we terminate exploration and restart the simulator. Different evaluated reward functions are presented in Table C.1 of Appendix C.

4.2.2 Double Deep Q-Networks

Two-step Learning Framework

First, we perform long-term generalised training to obtain a population model for the hormone delivery strategies. We use dilated recurrent neural networks (DRNNs) [128] for modeling the multi-dimensional time series including glucose levels, hormone doses, and meal intake. Other inputs affecting glucose levels, such as physical exercise, could also be considered [232]. To train the model, each basal hormone delivery (at five-minute intervals) is regarded as an action taken by the agent, while the glucose level on the next time step is set to the reward by the criteria of time in range (Equation 4.3). Secondly, by initialising the weights obtained from the population model, we have a model with good initial performance. With a TL process, we individualise the DQNs according to personal characteristics and safety constraints with a small subject-specific dataset. Safety constraints in the AP refer to a set of safety measures based on the observations by monitoring systems (e.g., CGM measurements), estimation of the metabolic state of the subject (e.g. IOB), and meal ingestion, to prevent or mitigate possible harmful BG events [233]. A safety supervision system can comprise multiple safety constraints tasked with potentially dangerous events that may arise in a clinical setting (e.g. manual inputs constraints, glucose sensor saturations, insulin and glucagon delivery limits).

During clinical trials, the data for training is usually very limited, thus we aim at fast learning performance. Therefore, we use a double DQN with modified importance sampling to further optimise approximated action values. A classic technique is employed to accelerate learning processes, where prioritised experience replay samples important transitions more frequently [234, 235]. To avoid overestimating the action values, a double DQN decouples action selection and value evaluation by two separate neural networks [236], as shown in Figure 4.3. The second step is suitable for a clinical trial setting, where the model is able to be fine-tuned in a relatively short period of time.

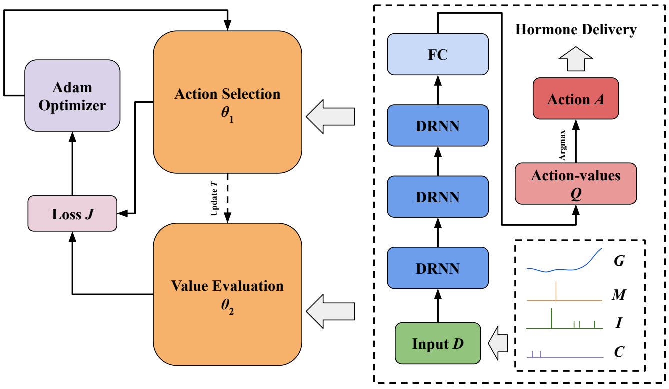


Figure 4.3: Diagram of the propose double DQN. The structure of the neural network is the same for both action selection and value evaluation, which consists of an input layer, a stack of DRNN layers, a fully-connected (FC) layer and output. The input data includes BG series from CGM G, meal M, insulin I and glucagon C.

Generalised DQN Training

In the first step, we use the simulator to generate an environment by using an average T1D subject for each one of the virtual cohorts (i.e. adult and adolescent). Compared with standard RNNs, DRNNs are preferred as DQNs for learning the delivery strategies. The large receptive field brought by dilation is powerful to extract features from glucose time series [128], where the dilated skip connection can be represented as

$$c_t^{(l)} = f\left(n_t^{(l)}, c_{t-d^{(l)}}^{(l)}\right), \tag{4.4}$$

where $c_t^{(l)}$ is the cell in layer l at time t, $n_t^{(l)}$ is the input to layer l at time t, $d^{(l)}$ denotes the dilation of layer l, and $f(\cdot)$ represents the output function of RNN cells. As shown in Figure 4.3, we use three DRNN layers with exponentially increasing dilation, to process the multi-dimensional time-aligned sequence and extract high-level features. Then training is carried out in the simulator with double DQN weights θ_1, θ_2, where action selections θ_1 and value evaluations θ_2 are obtained from two separate neural networks. According to Equation (4.2), the action-selection networks are trained with the loss as

$$\begin{aligned} J_{DQ}(Q) = \mathbb{E}_{(o,a,r,o')\sim\rho}[(r+ \\ \gamma Q(o',a';\theta_2) - Q(o,a;\theta_1))^2], \end{aligned} \tag{4.5}$$

where ρ is a mini-batch with transitions (o, a, r, o') sampled from the memory pool, and $a' = \arg\max'_a Q(o', a'; \theta_1)$ is chosen by the action selection DQN in Figure 4.3. Thus, the Q-function can be updated as

$$\begin{aligned} Q_{\theta_1}(o, a) \leftarrow Q_{\theta_1}(o, a) + \alpha(r+ \\ \gamma Q_{\theta_2}(o', \arg\max_a Q_{\theta_1}(o', a)) - Q_{\theta_1}(o, a)), \end{aligned} \tag{4.6}$$

where α is the learning rate, and the weights of θ_1 are copied to θ_2 with a fixed period. We optimise the learning rate by the Adam method at each iteration [155]. The corresponding pseudo-code is presented in Algorithm 1.

Algorithm 1 Generalised DQN Training

```
Input: the environment $E$ with average T1D subject parameters $Is$ provided by the
    simulator, update period $T_G$, $\varepsilon$-greedy
Initialise DQNs with random weights $\theta_1$, $\theta_2$, replay memory $\mathcal{B}$
for steps t$\in 1, 2, ..k$ do
    Sample action from $a \sim \pi(Q_{\theta_1}, \varepsilon)$, observe $o'$ in $E_{Is}$, calculate $r$, store $(o, a, r, o')$
    into $\mathcal{B}$
end for
repeat
    Sample action from $a \sim \pi(Q_{\theta_1}, \varepsilon)$, observe $o'$ in $E_{Is}$, calculate $r$, store $(o, a, r, o')$
    into $\mathcal{B}$
    Sample a mini-batch uniformly from $\mathcal{B}$ and calculate loss $J_{DQ}(Q)$
    Perform a gradient descent to update $\theta_1$
    if $t$ mod $T_G = 0$ then $\theta_2 \leftarrow \theta_1$ end if
until converge
```

For each meal, a standard dose of bolus insulin is delivered, and the agent explores random hormone delivery actions (single or dual) under policy π that is ε-greedy with respect to Q_{θ_1}. Human intervention could reduce training time and improve initial performance, but it would cause potential bias during the training process [237]. For *in silico* trials, random actions are tested with great flexibility and no safety concerns, as a great advantage of using a simulation environment. In this case, we can train the agent for a long time until the loss converges, so human intervention is not necessary. At the end of the generalised training, a population model consisting of a double DQN with weights θ_1 and θ_2 is obtained.

Personalised DQN Training

After developing a generalised model, we fine-tune the model by TL with regards to personal characteristics. We fetch the weights and features from the generalised model, then train the personalised DQNs within a dataset corresponding to a short period of time with safety constraints. We can choose to fine-tune all layers of the generalised model or to retain the weights of some of the lower layers and only fine-tune a higher-level portion of the network to avoid over-fitting. In experiments, we found that lower layers contain more generic features (e.g. insulin suspension during the trend of hypoglycaemia) that should be useful for all the subjects with T1D.

Here a method modified from [235] is used for calculating the loss of policy-generated data. Specifically, $J_n(Q)$ has an n-step returns (n= 12) to propagate values of actions to earlier states $r_i + \gamma r_{i+1} + \cdots + \gamma^{n-1} r_{i+n-1} + \max_a Q(o_{i+n}, a)$, and $J_{L_2}(Q)$ is an L2 regularization loss applied to θ to mitigate over-fitting. Prioritised experience replay samples the transitions with a probability Pr_i proportional to its importance priority [234], which is computed from previous data and normalised afterwards,

$$Pr_i = \frac{p_i^\alpha}{\sum_i p_i^\alpha}, \quad p_i = |\delta_i| + \epsilon', \tag{4.7}$$

where $\alpha \in [0, 1]$ determines the level of using prioritization, p_i is the priority of transition i calculated from last temporal difference (TD) error δ_i and ϵ' is a small positive constant. It allows the DQN to more frequently replay transitions with higher TD error. In addition, to ensure that hormones are delivered safely in the clinical trial, constraints $\mathcal{C}$ are applied to the suggested action before execution. Here we use a simple strategy for the safety constraints: suspending basal insulin or glucagon when the current BG level is below 80 mg/dL or over 160 mg/dL, respectively. In practice, the trend and prediction of BG levels can also be used in the safety constraints for early interventions. With proper training of the generalised model and adequate safety constraints, this algorithm can be adopted in a clinical trial setting. The corresponding pseudo-code detailing the algorithm is presented in Algorithm 2.

Algorithm 2 Personalised DQN Training

```
Input: replay memory $\mathcal{B}$ and DQNs weights $\theta_1'$, $\theta_2'$ from generalised training; individual environment $E$, safety constraints $\mathcal{C}$, update period $T_P$, parameter $\lambda_1, \lambda_2$,
Initialise personalised DQNs weights $\theta_1 \leftarrow \theta_1'$, $\theta_2 \leftarrow \theta_2'$
Initialise replay memory $\mathcal{D}$, merging $\mathcal{B}$ with priorities
for steps i$\in 1, 2, ..N$ do
    Sample action from policy $a \sim \pi(Q_{\theta_1})$,
    if $a$ subject to $\mathcal{C}$ then execute $a$ end if
    Observe $(o', r)$ in $E$
    Store $(o, a, r, o')$ in $\mathcal{D}$, overwriting the samples previously merged from $\mathcal{B}$
    Sample a mini-batch from $\mathcal{D}$ by modified importance sampling $Pr$ and update the transition priority
    Calculate loss $J(Q) = J_{DQ}(Q) + \lambda_1 J_n(Q) + \lambda_2 J_{L_2}(Q)$
    Perform a gradient descent to update $\theta_1$
    if $t$ mod $T_P = 0$ then $\theta_2 \leftarrow \theta_1$ end if
end for
```

4.2.3 In Silico Environment and Baseline

Following the architecture evaluation setup depicted in Figure 4.1, we conducted experiments to evaluate, *in silico*, the effectiveness of proposed DRL framework with the UVA/Padova T1D Simulator [81]. As stated in Section 4.2.1, we use two settings of control actions in the proposed DRL algorithm: single-hormone (DRL-SH) and dual-hormone delivery (DRL-DH). Following a TL strategy, we started with a long-term exploration with 1,500 simulated days to obtain a stable generalised model using Algorithm 1, then, we performed personalised training for each individual in the cohort (i.e. adult and adolescent) with 30 simulated days using Algorithm 2. Due to the significant amount of data required, the generalised model is meant to be trained in the simulator, whereas the personalised model training has the potential to be done in a clinical setting. Finally, the personalised models were tested in a period of 90 days. The details of DNN selection and hyperparameters are presented in Figure C.1 and Table C.2 of Appendix C, respectively.

The UVA/Padova T1D simulator provides an interactive environment for the agent to explore and learn the policy. We introduced additional intra-subject variability in the meal protocol scenario and the parameters of the T1D model [216]. In particular, we selected four meals as the daily pattern (average cases: 7 am (70 g), 10 am (30 g), 2 pm (110 g), 9 pm (90 g)) with meal-time variability (STD = 60 min) and meal-size variability

(CV = 10%). The meal-duration was set to 15 minutes. A misestimation of carbohydrate amount between −30% and +10% with uniform distribution was applied. The reason we used this skewed distribution is that the underestimation of carbohydrate content is more common than overestimation in real-life conditions, according to a cross-sectional study with 50 T1D subjects [130]. Variability for meal absorption and carbohydrate bioavailability were set to 30% and 10%, respectively. The variability of insulin sensitivity was considered to be 30% for adult cohort and 20% for adolescent cohort, which are created by the scenario function in the subjects' own profile. These values of variability were selected based on available physiological knowledge and to achieve the glycaemic outcomes commonly observed in such populations when treated with standard therapy [238]. We saved intra-day and intra-person variability for each subject and used the same scenarios for all the evaluated methods, i.e. same daily events and variability time series, in order to have a fair comparison. We utilised the 10 virtual adults and 10 virtual adolescents, plus the corresponding average subjects, for generalised training.

As a baseline method, a low-glucose insulin suspension (LGS) strategy, commonly found in SAPs, was employed [189]. LGS systems have been proven to reduce hypoglycaemia by suspending basal insulin delivery [134]. For meal bolus calculation, a SBC was used [228].

4.2.4 Results and Discussion

To evaluate the performance of the proposed algorithms and compare them against the baseline method, we selected five standard glycaemic metrics commonly employed by the diabetes technology community [239]. These include: percentage time in the glucose target range of $[70, 180]$ mg/dL (TIR), percentage time below range (TBR) (i.e. hypoglycaemia), percentage time above range (TAR) (i.e. hyperglycaemia), mean BG levels, and risk index (RI). Results are expressed by mean values and standard deviations (Mean±STD).

Table 4.1 and Table 4.2 shows the results of the three tested methods evaluated on the adult and adolescent cohorts, respectively. Compared with LGS therapy, both single-hormone and dual-hormone DRL models improve the glucose control performance

Table 4.1: Testing performance of glucose control on the adult virtual cohort

Method	TIR (%)	TBR (%)	TAR (%)	Mean BG	RI
LGS	77.55±6.78	2.87±1.38	19.58±5.79	140.78±8.23	2.52±0.89
DRL-SH	80.94±7.00*	2.06±1.33*	17.00±5.82	140.36±5.98	2.28±0.72
DRL-DH	85.55±7.33**,†	1.92±1.90*	13.81±6.67**,†	140.12±8.13	2.16±0.65†

Symbol * and † indicates statistical significance ($p < 0.05$) with with respect to LGS and DRL-SH, respectively. A double symbol (e.g. ‡) indicates statistical significance ($p < 0.01$).

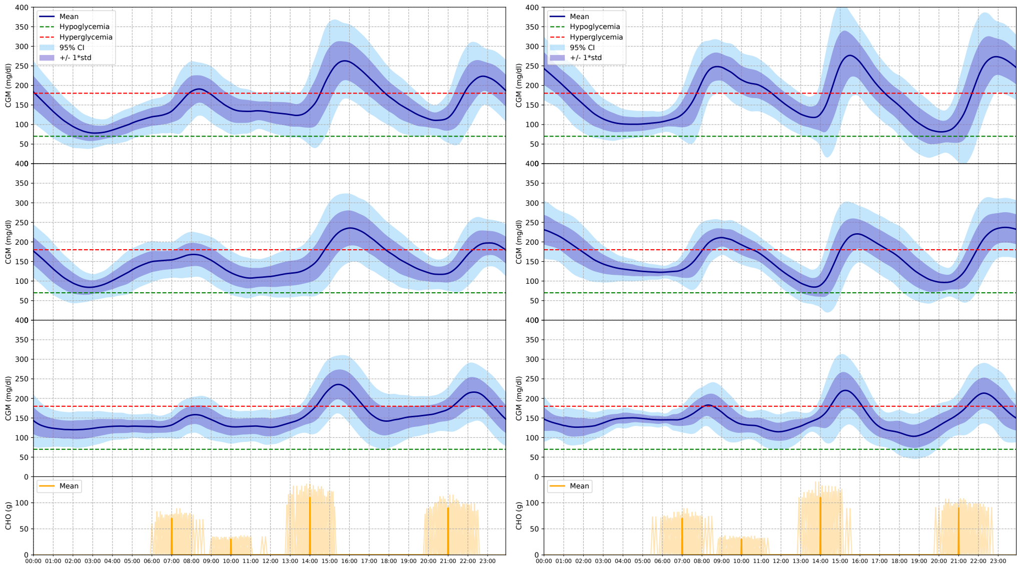
Table 4.2: Testing performance of glucose control on the adolescent virtual cohort

Method	TIR (%)	TBR (%)	TAR (%)	Mean BG	RI
LGS	55.50±14.68	6.93±4.69	37.57±11.64	162.15±20.46	4.76±2.70
DRL-SH	65.85±16.30**	5.51±3.37	28.63±14.36**	151.18±18.26**	3.99±2.43**
DRL-DH	78.83±6.60**,†	2.64±1.96**,‡	18.53±6.48**,†	149.96±8.83**	2.94±0.99**,‡

Symbol * and † indicates statistical significance ($p < 0.05$) with with respect to LGS and DRL-SH, respectively. A double symbol (e.g. ‡) indicates statistical significance ($p < 0.01$).

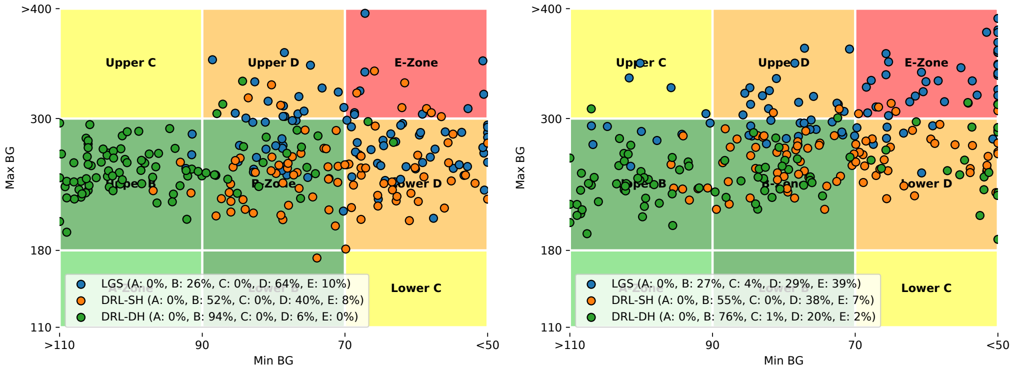
by reducing hypoglycaemia, hyperglycaemia and increasing TIR in the two cohorts. Of note, the dual-hormone DRL model significantly increases the mean TIR with a notable decrease of risk index, achieving the best performance. Mean BG levels are maintained in the adult cohort, while the improvement is significant in the adolescent cohort.

For demonstration purposes, Figure 4.4 graphically displays the performance of the three evaluated methods for a chosen adult and a chosen adolescent over a three-month testing period. In particular, the glucose profile over 24 hours (Mean±STD) (i.e. ambulatory glucose profile) is presented. The displayed results in Figure 4.4 are consistent with the numerical results corresponding to the overall population presented in Tables 4.1 and 4.2. Control variability grid analysis (CVGA) is a commonly used tool for evaluating closed-loop insulin delivery techniques [240], as shown in Figure 4.5. It plots the extreme (minimum/maximum) BG values on a grid with 9 zones, which has been widely used to compare the efficacy of different algorithms for *in silico* and clinical trials. The points in A+B zones stand for optimal glycaemic control. It is worth noting the significant improvement achieved by DRL-DH when compared with LGS. In particular, the percentage of points in the A+B zones increases from 26% to 94% for the adult cohort and from 27% to 76% for the adolescent cohort.



(a) (b)

Figure 4.4: Performance of the three methods on an adult subject (a) and an adolescent subject (b) over the three-month testing period: (Top-to-bottom) LGS, DRL-SH, DRL-DH, carbohydrate distribution. The average BG levels are shown in solid blue lines, and the hypo- and hyperglycaemia thresholds are shown in dotted green and red lines, respectively. Blue shaded regions show the 95% CI, and the purple shaded regions indicate the standard deviation.



(a) (b)

Figure 4.5: CVGA plot for the adult (a) and the adolescent (b). The blue, orange and green dots represent the LGS, DRL-SH and DRL-DH results, respectively.

It is observed that there is a significant difference between results for adults and adolescents. This is because the simulator used different metabolic and demographic parameters for the two cohorts and set a higher glucose variability in adolescents than in adults [81]. Adolescents usually have lower body weight and higher insulin sensitivity.

To the best of our knowledge, this is the first study that systematically evaluates, *in silico*, a DRL algorithm to control BG levels with single-hormone and dual-hormone delivery, using the latest T1D simulator (S2013) [81] and additional intra-subject variability. In the presented *in silico* experiments, when compared against the LGS baseline method, the proposed DRL model achieves superior performance in terms of glycaemic outcomes. Comparing the proposed model with existing closed-loop insulin delivery techniques, although interesting, is a challenging task due to the difficulty in replicating the testing scenarios and the tuning of the controllers. Hence, a head-to-head comparison is not performed. Although not directly comparable, an informal comparison with existing works in the literature on reinforcement learning for insulin and glucagon delivery has been done. In [206], the authors propose an reinforcement learning controller and achieve the adult TIR of 89% on the UVA/Padova simulator, which is close to the performance achieved by our DRL-DH model. In this previous work, both basal and bolus insulin delivery are optimised, while in our work only basal insulin delivery is optimised using different variability in the simulator. In [241], Ngo *et al.* use reinforcement learning to optimise control parameters in glycaemic models without providing comparable TIR results. In [102], the authors propose a DQNs algorithm to control single hormone (insulin) delivery and they evaluate it on the previous version of the UVA/Padova simulator (S2008). However, no comparable glycaemic outcomes are provided. Therefore, our work not only proposes a novel DRL algorithm for insulin and glucagon delivery but also serves as a benchmark for the future evaluation of other control algorithms. Future work includes comparing the proposed DRL model with classic control algorithms in AP systems [242], such as model predictive controller [139, 191, 192], proportional-integral-derivative [233], and fuzzy logic [194, 243]. In this work, a total of six discrete actions are considered in a dual-hormone AP system. Smoother actions or continuous action space could be considered in future work to further improve glycaemic control. This feature is particularly important for meal insulin bolus delivery as dietary intake varies widely. Therefore, we adopt an actor-critic model with continuous action space in the next section.

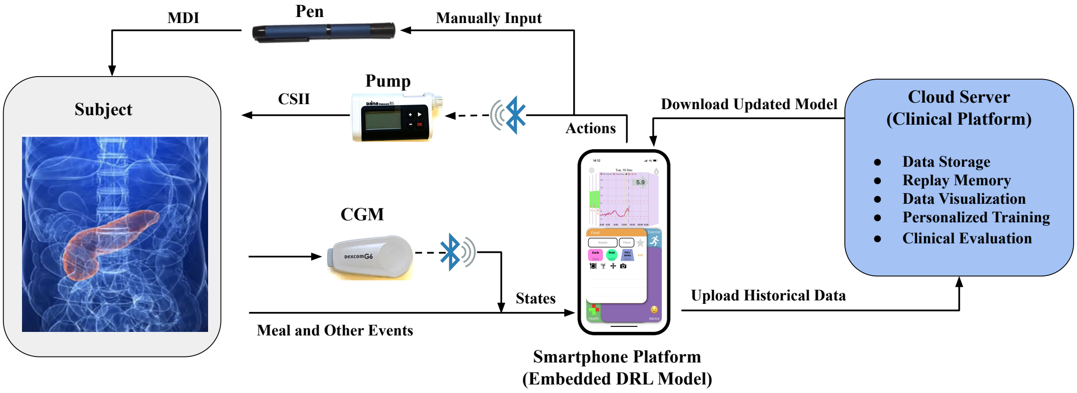


Figure 4.6: System architecture to evaluate the DRL models for meal insulin bolus recommendation in an ambulatory clinical setting.

4.3 Meal Insulin Bolus Recommendation

This section presents a novel DRL advisor for meal insulin bolus recommendation in T1D management. Figure 4.6 shows the system architecture to incorporate the DRL advisor in a smartphone app that collects real-time data from two sources. One is wearable sensors for acquiring physiological data via Bluetooth, and the other is a manual log for recording any exogenous events, such as meals, exercise, and health conditions. Based on an input state, the embedded DRL advisor can calculate a corresponding action, i.e., the gain of meal insulin bolus, to assist users to control the insulin pump with CSII, or insulin pen with MDI. The historical data is automatically uploaded to a cloud server for monitoring and backup purposes. Moreover, the uploaded data forms a pool of replay memory, where the DRL models can be further updated using new data and personalised training. Finally, the app automatically fetches the updated model and saves it in local storage.

4.3.1 Problem Formulation

We consider a standard DRL setup to formulate the problem of insulin bolus advisor, which employs an agent to deliver insulin and interact with the environment of diabetes E_D in discrete timesteps. At timestep t, i.e., meal time, the agent first receives an observation s_t from the sensing devices in the glucose control system and takes an action a_t to deliver a certain amount of insulin bolus. Then the physiological state of the T1D subject transits to

s_{t+1} and returns the reward r_{t+1} based on the evaluation of postprandial glucose excursion. In this regard, this problem can be modeled as a Markov decision process with a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, R \rangle$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is action space, P is the transition functions between states and R is a reward function. The policy of the agent maps the distribution of actions for a given state, which is defined by $\pi : \mathcal{S} \rightarrow \mathcal{P}(\mathcal{A})$. Under the policy π, the action-value function $Q^{\pi}(s_t, a_t)$ is the sum of the discounted future reward with the current state s_t and chosen action a_t. The target of reinforcement learning is to obtain the optimal policy to gain maximum reward. Moreover, a policy μ can be modeled by mapping the state to a specific action in a deterministic environment as in our case: $\mu : \mathcal{S} \rightarrow \mathcal{A}$. In this regard, the estimated target policy can be solved by the Bellman equation:

$$Q^{\mu}(s_t, a_t) = \mathbb{E}_{r_{t+1}, s_{t+1} \sim E_D} \left[R(s_t, a_t) + \gamma Q^{\mu}(s_{t+1}, \mu(s_{t+1})) \right], \tag{4.8}$$

where γ is a discount factor within the range of $[0, 1]$, and Q^{μ} can be learned by off-policy. In the context of glycaemic control, the input and output of the algorithm are the current observation of physiological states and insulin bolus suggestion, respectively. The objective is to obtain the meal insulin bolus that optimises glycaemic control by solving Equation (4.8). Safety constraints on insulin suggestions are required to avoid undesirable glucose events in a clinical setting.

In this problem, the insulin bolus varies largely depending on the meal ingestion and will significantly influence the postprandial BG levels. It is difficult to discretize the range of feasible bolus insulin doses as the action set with value-based DRL in Section 4.2.1. Too many intervals will exponentially slow the training process, while a small action set can degrade the performance due to the variability of meals. Alternatively, we could get an estimation of the dosage with a bolus calculator and then vary a continuous range (e.g., ±30%) around this value. Therefore, we introduce an actor-critic method, deep deterministic policy gradient (DDPG) [244], to enable a continuous action-space for the agent-based on deterministic policy gradient [245]. The critic function $Q(s, a)$ is recursively learned by the Bellman equation in Equation (4.8). With the initial distribution J of the parameters, the actor function $\mu(s)$ is updated as:

$$\nabla_{\theta^\mu} J \approx \mathbb{E}_{s_t \sim \rho}[\nabla_a Q(s, a|\theta^Q)|_{s=s_t, a=\mu(s_t)} \nabla_{\theta^\mu} \mu(s|\theta^\mu)|_{s=s_t}], \tag{4.9}$$

where θ^Q and θ^μ are the parameters of critic and actor, respectively, ρ is the state-visitation distribution. Specifically, the actor decides how many units of bolus insulin to deliver for the current physiological state, whilst the critic determines how good the action was taken and tells the actor how to adjust.

Agent States And Actions

For the SBC in AP systems with CGM and insulin pumps [246], a relatively empirical formula is used to calculate the insulin dose:

$$Bolus_t^* = \frac{CHO_t}{ICR} + \frac{G_t - G^T}{ISF} - IOB_t, \tag{4.10}$$

where CHO_t is the total carbohydrate amount of the meal ingestion (gram), ICR is the insulin-to-carbohydrate ratio (g/IU); ISF is the insulin sensitivity factor (mg/l/IU); G_t is the current reading of BG level from CGM (mg/dl); G^T is the target BG level; and IOB is insulin on board. The ISF is commonly multiplied by a portion to adapt correction insulin [228]. IOB can be estimated by the previous bolus dose using various methods, e.g., dynamic rule-based algorithms [247], ANNs [248], circadian insulin sensitivity variation [249], or a simple formula as:

$$IOB_t = Bolus_{t-1} * \max(0, (1 - \frac{(ts_t - ts_{t-1})}{T_{IOB}}), \tag{4.11}$$

where ts is the time of bolus delivery, and T_{IOB} is a manually defined interval to indicate the active time. ts stands for the time sampled by the CGM at fixed intervals, e.g., every 5 min. In real clinical practice, these parameters $\{ICR, ISF, T_{IOB}\}$ are not time-varying and might depend on the physiological state of the subject. This underline non-linear function can be approximated by the actor function with the weights of DNNs $\mu(\theta^\mu)$. Furthermore, considering DNNs have the superior capability of representation learning [18], we extend the G_t into a series of historical record of CGM $\mathbf{G}_t$ to extract more features from input states, such as the BG trends. As a result, the agent state at timestep t is denoted as:

$$s_t = \{\mathbf{G}_t, CHO_t, ts_t, IOB_t\} \in \mathcal{S}, \tag{4.12}$$

where $\mathbf{G}_t \in \mathbb{R}^{1\times L}$ contains a number (L) of BG measurements from CGM.

In most cases, the action space of DDPG methods is defined by a range. The range of insulin bolus could be relatively large due to the uncertainty of meal carbohydrate intake. To improve the convergence of the training, we define the bolus actions from DDPG as:

$$Bolus_t = [\frac{CHO_t}{ICR}, \frac{G_t - G^T}{ISF}, -IOB_t] * \mu(s_t|\theta^\mu)^T \in \mathcal{A} \tag{4.13}$$

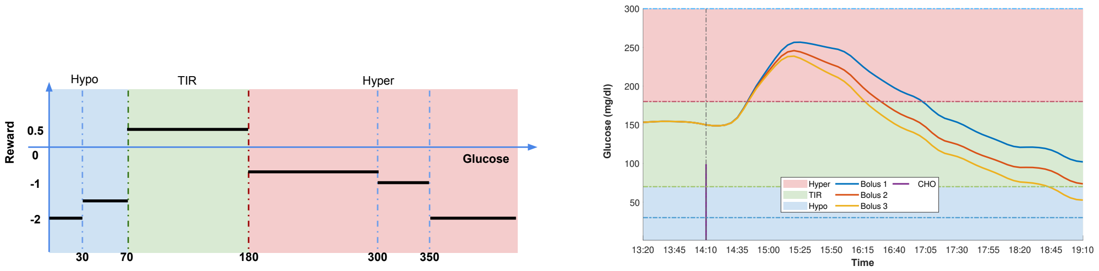
where the output of the actor function $\mu(s_t|\theta^\mu) = [g_{ICR}, g_{ISF}, g_{IOB}]$ is defined as a three-element vector, consisting of the gains to respectively adapt ICR, ISF, and IOB for the SBC. In this case, the range of the gains is defined as [0.2, 2], to reduce or amplify the bolus insulin. The action is the same as the SBC when $\mu(s_t|\theta^\mu) = [1, 1, 1]$.

Reward Function

The goal of a bolus advisor is to maximise postprandial BG levels in the target zone, i.e., [70, 180] mg/dL while minimising the the occurrence of hypoglycaemia [239, 250]. To guide the agent to achieve this goal, the positive rewards are applied to the TIR zone. Employing CGM systems, we obtain a series of postprandial BG levels with a fixed sampling period, which allows us to assign a reward value for each postprandial BG reading then sum them up:

$$r_{t+1} = \frac{1}{ts* - ts_t} \sum_{k=ts_t}^{ts*} f_R(G_k), \tag{4.14}$$

where $ts* = \min(ts_{t+5h}, ts_{t+1})$. If the time interval between two successive states (meal ingestion) is too large, we only consider 5-hour postprandial period after the current



(a) (b)

Figure 4.7: Illustration of proposed reward function to determine the performance of the action that was taken. (a) Step function to calculate the reward for the discrete BG values after dietary intake, referring to Equation (4.15). The blue, green, and red regions stand for hypoglycaemia, euglycaemia, and hyperglycaemia zones, respectively; (b) Postprandial glucose curves corresponding to three different bolus and same variability.

meal [250].The discrete reward is formulated as:

$$f_R(G_k) = \begin{cases} 0.5, & 70 \leq G_k \leq 180, \\ -0.8, & 180 < G_k \leq 300, \\ -1, & 300 < G_k \leq 350, \\ -1.5, & 30 \leq G_k < 70 \\ -2, & \text{else}. \end{cases} \tag{4.15}$$

Figure 4.7a depicts the proposed reward function. Figure 4.7b shows a comparison of the postprandial excursions corresponding to three different insulin bolus $\{Bolus_1, Bolus_2, Bolus_3\}$ for the same meal and variability. The TIR and rewards for the three insulin boluses are {53.3%, 65.0%, 58.3%} and {−0.107, 0.045, −0.112}, respectively. Although $Bolus_3$ obtains better TIR than $Bolus_1$, its reward is smaller than that of $Bolus_1$, which is due to the increase of hypoglycaemia. People with hypoglycaemia episodes are at major risk of acute short-term complications (e.g., coma), which is in general less preferable than hyperglycaemia [25].

4.3.2 Deep Deterministic Policy Gradient

To learn a generalised policy in a large action-state space, we use DMLP as the non-linear approximators to parameterise the actor and critic functions in DDPG, which means θ^Q and θ^μ in Equation (4.9) become the weights of neural networks. Following the success of the DRL on human-level control [80], we employ replay memory $\mathcal{M}$ and fixed target networks to further improve the stability of the proposed methods. $\mathcal{M}$ stores last N transitions with a tuple $\langle s_t, a_t, r_{t+1}, s_{t+1}\rangle$ to provide experience samples for off-policy learning. Using fixed target networks, we have separated DNNs to calculate targets during the model training. Thus, the target neural networks of actor Q' and critic μ' are obtained by copying weights of current actor Q and critic μ with a fixed period $\mathcal{T}$, using soft update [244]. Referring to Bellman Equation (4.8) and transition tuples, the loss of critic is formulated as follows:

$$\mathcal{L}(\theta^Q) = \mathbb{E}_{s_t \sim \rho}[(r_{t+1} + \gamma Q'(s_{t+1}, u'(s_{t+1}|\theta^{u'})|\theta^{Q'}) - Q(s_t, a_t|\theta^Q)^2]. \tag{4.16}$$

Particularly, the value-based algorithm includes calculating the TD error to update the weights of critic neural network with Equation (4.16), while the policy-based part is using the outcomes of critic and Equation (4.9) to update actor neural network. Repeatedly updating the weights of critic and actor DNNs, the model can learn the policy to maximise the expected reward by delivering optimal bolus, as depicted in Figure 4.8. We instantiate DNNs with three fully connected hidden layers for both the actor and critic.

4.3.3 Two-Step Learning Framework

Collecting large sets of clinical data is often expensive, and evaluating algorithms on human subjects without pre-clinical validation, or proper safety constraints, might be dangerous. To this end, we extend the use of the two-step learning framework described in Section 4.2.2 and employ the UVA/Padova T1D simulator. In particular, at the first step, the agent is allowed to explore random actions by adding Gaussian noise $\mathcal{N}(0, 0.3)$ and constantly interact with the simulator. For this purpose, we use average T1D subjects provided by

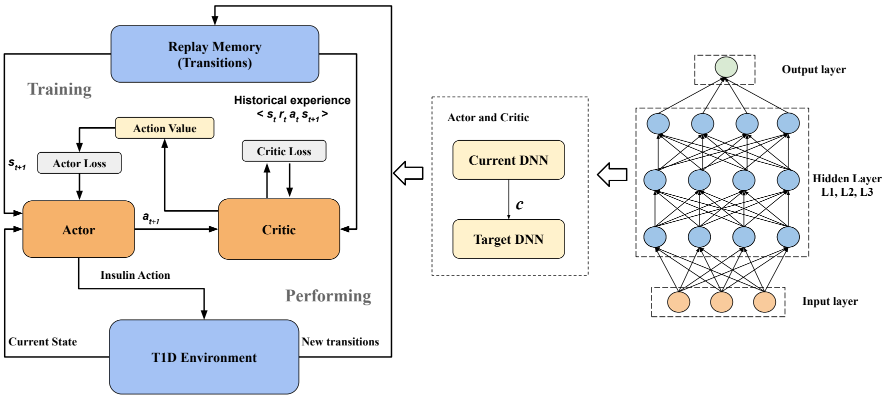


Figure 4.8: The block diagram of the proposed DDPG model with the actor-critic architecture.

the simulator. To obtain a population DRL model, the agent performs long-term off-policy learning until the loss of critic converges. In the second step, the weights of the population model are used to initialise a personalised model for each individual by TL. Then the models are further fined-tuned by using subject-specific data with safety constraints in a short-term training period of T_p. Here we use a simple constraint during in silico validation: the action gain is limited to be greater than one if current BG enters hyperglycaemia and less than one for hypoglycaemia. In a clinic practice, more advanced constraints could be used, such as a interval arithmetic-based dynamic insulin constraint proposed by Liu *et al.* [189]. During the fine-tuning, the data is collected with a form of transitions, i.e., a tuple of $\langle s_t, a_t, r_{t+1}, s_{t+1} \rangle$. According to Equation (4.12), such a transition requires multiple data fields, including CGM measurements, estimated carbohydrate of meal ingestion, mealtime, and dosages of insulin bolus. These data fields are available with the proposed system architecture in Figure 4.6, where carbohydrate estimation is manually entered, while other data can be collected automatically. Finally, we test the personalised models on separate testing sets.

To accelerate the training process, we adopt a variant of prioritised memory replay to sample mini-batches of the transitions [234]. The priority, i.e., the probability of sampling a transition $Pr(i)$, is based on the magnitude of the TD error δ_i, which is denoted

Algorithm 3 DDPG Insulin Bolus Advisor

```
**Input:** average environment $E_a$, individual environment $E_i$, safety constraints $\mathcal{C}$, update period $\mathcal{T}$.
**if** personalised training **then**
    Initialise the weights $\theta^Q$, $\theta^\mu$ from the population model, $\mathcal{N} = 0$, $E_D = E_i$,
**else**
    Randomly initialise the weights $\theta^Q$, $\theta^\mu$ for actor $Q$ and critic $\mu$, $\mathcal{C} = \phi$, $E_D = E_a$
**end if**
Copy the weights to the target networks $Q'$ and $\mu'$ : $\theta^{Q'} \leftarrow \theta^Q$, $\theta^{\mu'} \leftarrow \theta^\mu$
Initialise empty replay memory $\mathcal{M}$ with the volume of $N$ and prioritization
**repeat**
    Observe state $s_t$ from $E$, select action by actor $a_t = \mu(s_t|\theta^\mu) + \mathcal{N}$,
    **if** $a_t$ subject to $\mathcal{C}$ **then** execute $a_t$ in $E$ **else** restrict $a_t$ by $\mathcal{C}$ **end if**
    Observe state $s_{t+1}$, calculate reward $r_{t+1}$, store the transition $(s_t, a_t, r_{t+1}, s_{t+1})$ in $\mathcal{M}$
    Sample a mini-batch from $\mathcal{M}$ by priority $Pr$
    Calculate the loss of critic $\mathcal{L}(\theta^Q)$ and update the weights $\theta^Q$ with importance weights $w$
    Calculate TD error, update $Pr$ and $w$
    Perform a gradient descent $\nabla_{\theta^\mu} J$to update $\theta^u$
    **if** $t$ mod $\mathcal{T} = 0$ **then** soft update: $\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'}$, $\theta^{\mu'} \leftarrow \theta^\mu + (1-\tau)\theta^{\mu'}$ **end if**
**until** the loss of critic converges or t = $T_p$
```

as:

$$Pr(i) = \frac{(|\delta_i| + \epsilon)^\alpha}{\sum_{n=1}^{N}(|\delta_n| + \epsilon)^\alpha}, \tag{4.17}$$

where N is the total number of transitions in replay memory; ϵ is a small positive constant to guarantee that the transitions with zero TD error can also be sampled; and α is the degree of using prioritization ($\alpha = 0$ stands for uniform sampling). To remove the bias of the prioritised sampling and improve convergence, a set of importance weights is introduced and normalised as: $w_i = (N * Pr(i))^{-\beta} / \max_n w_n$, where β is the degree to compensate the prioritization ($\beta = 1$ means full compensation). The details of the complete training algorithm are presented in Algorithm 3. The hyperparameters are listed in Table C.3 of Appendix C.

4.3.4 In Silico Validation

To validate the performance of the proposed algorithm, we use a customised version of the UVA/Padova T1D simulator [81] as the platform to conduct pre-clinical trials. For this purpose, we tested the models on 10 adult and 10 adolescent virtual subjects. Additional intra-day variability was introduced to better emulate real-life conditions [216]. In particular, we employ a daily pattern with three realistic meals: breakfast, lunch, and dinner. The time and carbohydrate content are as follows: 7 a.m. (70 g), 2 p.m. (110 g), and 9 p.m. (90 g), and the duration of each meal is set to 15 min. The variability of meal-time and meal-size are set to STD = 30 min and CV = 10%, respectively. In addition, we consider that the subjects are likely to under- or over-estimate the carbohydrate content of meals by 70% and 110%, respectively. The intra-subject insulin sensitivity variability is set to 30% following a sinusoidal function [216]. Although a single dose of basal insulin might not be optimal to cover the basal insulin requirements due to the presence of intra-subject variability, this is a common practice in people on MDI. Hence, we wanted to test the viability of our proposed approach on this subpopulation, which represents the majority of people with T1D.

In this work, we first use the average virtual subjects to train a generalised model over a long period until the performance, i.e., the learning curve, is stable. This step includes random exploration, which needs to be done in the simulator. With proper safety constraints and initialisation, personalised training can be then conducted in an actual clinical setting. Here, we perform the second training step on 180 simulated days (6 months) to fine-tune the personalised model in the simulator. This setting is determined by the convergence of the models, i.e., the learning curves. The computational time of model training is short, which is around 10 milliseconds for each step, but it takes a long time to collect transitions in the training sets, since there are only a few meal events (i.e., 3–4 transitions) per day. If waiting 6 months is considered too long to converge to an optimal performance in an actual clinical setting, it is possible to use a larger learning rate or stop training earlier. However, in this case, the model could achieve sub-optimal performance. Finally, we test the personalised model in a period of 90 days (3 months), and the new

Table 4.3: Glycaemic control metrics evaluating the performace of the DRL and SBC algorithms on the 10-adult virtual cohort. Statistical significance is indicated as † for $p \leq 0.01$.

Method	TIR (%)	TBR (%)	TAR (%)	Mean BG	CV (%)	LBGI	HBGI
SBC	74.1 ± 8.4	5.5 ± 1.9	20.2 ± 8.2	138.6 ± 11.5	34.8 ± 4.8	1.5 ± 0.5	4.1 ± 1.7
DRL	$80.9 \pm 6.9^\dagger$	$1.9 \pm 1.5^\dagger$	17.0 ± 6.1	138.1 ± 7.5	$31.1 \pm 5.3^\dagger$	$0.7 \pm 0.4^\dagger$	3.6 ± 1.2

simulations are generated by the same meal protocol. To make a fair comparison between the proposed model and baseline algorithms, the same scenarios and randomness seed of variability are saved and used for each evaluated method.

4.3.5 Results and Discussion

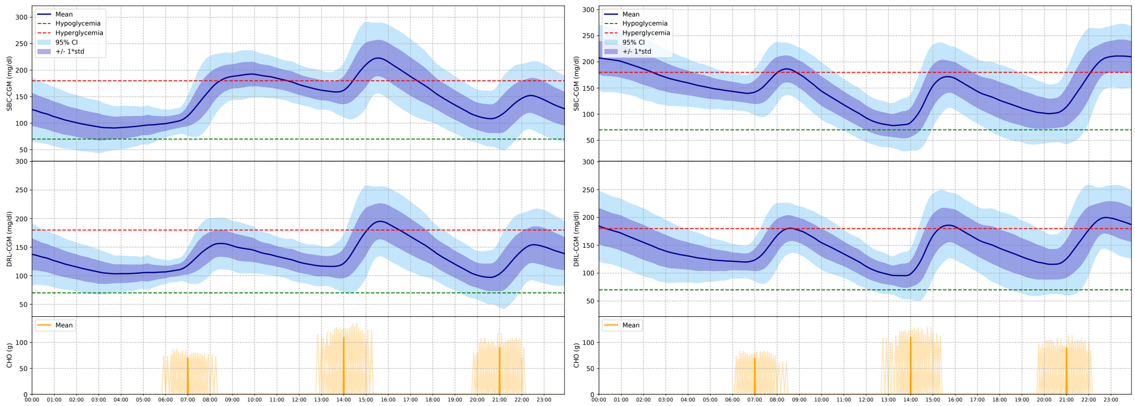
To measure the performance of glycaemic control, we use a set of commonly employed metrics, including TIR, TBR, and TAR. The mean BG values, CV, low blood glucose index (LBGI), and high blood glucose index (HBGI) are also used to present a comprehensive evaluation. Furthermore, CVGA [240] is employed to visualise the glycaemic outcomes.

In order to evaluate the performance of the proposed algorithm, we employed a baseline method consisting of the SBC with fixed parameters (Equation (4.10)) [228]. The results of personalised DRL models and the baseline method are presented as Mean±STD. In particular, we use the paired t-test to compute p-values to analyze the statistical significance, and the normality of data distribution is tested using histograms. Table 4.3 and 4.4 show the glycaemic outcomes for the adult cohort (n = 10) and adolescent cohort (n = 10), respectively, over 3 months. It is to be noted that the DRL algorithm achieves better performance than the SBC for every evaluated metric. The TIR results have been significantly enhanced for the adult and adolescent cohorts with a significant decrease in hypoglycaemia and hyperglycaemia. The mean BG level is improved for the adolescent cohort and maintained for the adult cohort. Finally, LBGI and HBGI, as the key metrics for measuring the risk of hypoglycaemia and hyperglycaemia, are largely improved.

Figure 4.9 shows an average glucose profile over a 24-hour period for two chosen adult and adolescent subjects to illustrate the improvement of DRL compared with the

Table 4.4: Glycaemic control metrics evaluating the performace of the DRL and SBC algorithms on the 10-adolescent virtual cohort. Statistical significance is indicated as * for $p \leq 0.05$ and † for $p \leq 0.01$.

Method	TIR (%)	TBR (%)	TAR (%)	Mean BG	CV (%)	LBGI	HBGI
SBC	54.9 ± 12.4	6.5 ± 3.5	38.5 ± 13.0	167.5 ± 25.3	40.7 ± 6.1	2.4 ± 1.7	9.2 ± 4.9
DRL	$61.6 \pm 14.1^\dagger$	$4.3 \pm 2.4^*$	$34.1 \pm 13.6^*$	161.6 ± 24.7	$38.6 \pm 7.5^*$	$1.3 \pm 0.8^*$	$8.0 \pm 4.8^*$

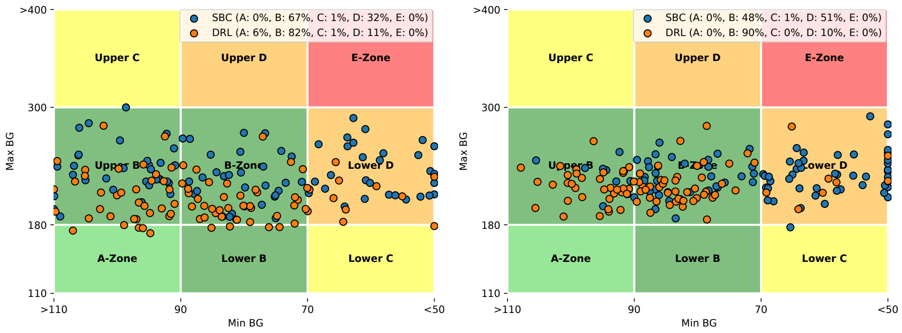


(a) An adult subject. (b) An adolescent subject.

Figure 4.9: Graphical example of the improvement on glycaemic control of the DRL algorithm over the SBC method. (a), (b) show the performance of an adult and adolescent subject, respectively. From top to bottom, each plot shows the daily glucose trajectory of SBC and DRL and distribution corresponding to three meal ingestion over 3 months. The thresholds of hyperglycaemia and hypoglycaemia are displayed in red and green dashed lines. The solid blue lines indicate the average BG levels. The blue and purple shades indicate the 95% CI and standard deviation, respectively.

SBC method. After learning the personalised strategies, the well-trained DRL agent delivers an optimal insulin bolus that effectively reduced the postprandial hyperglycaemia without increasing hypoglycaemia.

Figure 4.10 depicts the corresponding CVGA plots for the chosen adult and adolescent subjects. Here we customised the CVGA plots, where each dot stands for the glycaemic performance over 24 h, i.e., daily glucose trajectory of the same subject. The results, i.e., the distribution of the scattering dots, are consistent with the glucose profile in Figure 4.9. It is worth noting that, compared with the SBC method, the percentage improvement in the A+B zone increased from 67% to 88% for the adult subject and from 48% to 90% for the adolescent subject. The dot distribution of the DRL method shifted



(a) The adult subject.

(b) The adolescent subject.

Figure 4.10: CVGA plots comparing the SBC (orange dots) and DRL (blue dots) methods corresponding to a chosen adult subject (a) and a chosen adolescent subject (b) over a three-month scenario.

towards bottom-left corner, which is an indicator of good glycaemic control.

In this section, we proposed a novel algorithm for meal insulin bolus dosing based on DRL. To the best of our knowledge, this work is the first attempt employing DRL to develop personalised insulin bolus advisor in T1D. Although many pioneering studies have used the UVA/Padova T1D simulator to develop glycaemic control algorithms, the different settings in meal-protocols, variability, randomness in the scenarios make it challenging to perform a direct head-to-head comparison between the existing works. In addition, sometimes the existing algorithms are evaluated in combination with basal insulin control [206, 212]. Hence, we evaluated the proposed DRL algorithm with commonly employed metrics to comprehensively assess its performance. This data-driven algorithm also has the potential to be applicable to support people with T2D on insulin. However, this requires further study and will be the subject of future work.

The results presented in Tables 4.3 and 4.4 show that the DRL algorithm achieves good *in silico* glycaemic control on a customised version of the the FDA-accepted UVA/Padova T1D simulator. Compared with SBC, the proposed methods achieves a significant improvement in TIR and hypoglycaemia reduction for both adult and adolescent virtual cohorts. Using the system architecture in Figure 3.1, the proposed method can be implemented on a smartphone app and updated by the cloud server without much engineering work. Therefore, it suggests that the DRL algorithm has the potential to improve meal bolus insulin bolus delivery for T1D subjects in actual clinical settings.

To validate the robustness of the proposed method, we introduced additional intraday variability to the in silico trials and tested the algorithm on 20 different subjects. As depicted in Figure 4.9a, for the chosen adult, the DRL algorithm reduced postprandial hyperglycaemia after lunch, while the mean BG levels before dinner time remains above the hypoglycaemia threshold thanks to the strict reward setting in the low BG zone (Figure 4.7). However, the improvement by optimal insulin bolus is less significant in some extreme scenarios, e.g., a highly insulin-sensitive subject ingesting a meal with high carbohydrate content. Thus, adaptive control of basal insulin and glucagon (optional) by DRL would be helpful in these cases in T1D subjects wearing insulin pumps [251], as described in Section 4.2. Although we introduced variability into carbohydrate misestimation, it might be worth evaluating separately the effects of under- or overestimation of carbohydrate content and investigating how these errors will influence the final strategies by the proposed DRL model.

4.4 Offline DRL for Basal Insulin Control

In this section, we describe the problem of basal insulin control using the framework of offline DRL and OPE, as shown in Figure 4.11, which includes a total of four steps. The first step is to train offline DRL agents with different hyperparameter settings to obtain multiple policies. Then we train a value function for each learned policy with the OPE method and the same training data. Next, we use the trained OPE to estimate policy values on validation data and select the best policy with the highest value. The final step is to evaluate the selected policy on testing data and/or in clinical trials to demonstrate unbiased performance.

4.4.1 Problem Formulation

The DRL environment of glucose control can be modelled as a Markov decision process, denoted by a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, r, \gamma, \rho_0)$, where $\mathcal{S}$ denotes the state space; $\mathcal{A}$ is a set

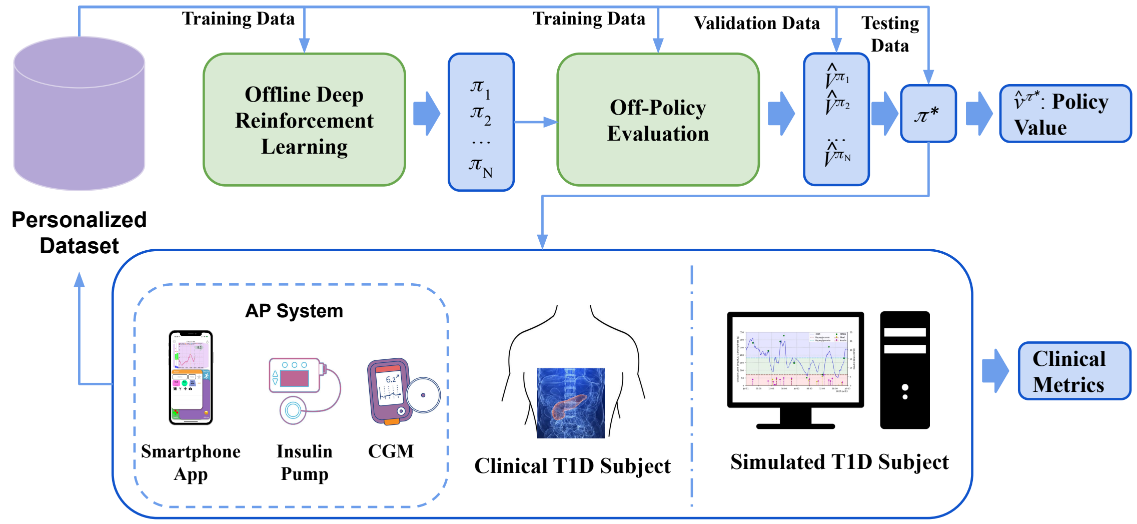


Figure 4.11: System architecture of developing glucose control algorithms in T1D management using the proposed offline DRL and OPE framework. The thin and thick arrows indicate the input and output of each module, respectively. The model training, validation, and testing are performed in completely offline settings. The clinical trials or *in silico* simulation can be further conducted to obtain clinical metrics.

of actions; $\mathcal{T}(s_{t+1}|s_t, a_t)$ defines the transition distribution; $r : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ is the reward function; $\gamma \in (0, 1]$ is a discount factor, and ρ_0 represents the distribution of initial states.

In general, the goal of DRL is to optimise the performance of a policy $\pi(a_t|s_t)$ in a given environment or on a historical dataset $\mathcal{D}$ in terms of offline settings. Typically, such an offline dataset is collected from one or multiple behaviour policy $\pi_\beta(a_t|s_t)$, which may be different from $\pi(a_t|s_t)$, and contains a set of historical transitions $\mathcal{D} = \{s_t^i, a_t^i, s_{t+1}^i, r_{t+1}^i\}$. Denoting an episode (i.e., a trajectory) τ with a length of L, the dataset can also be defined as $\mathcal{D} = \{\tau^i\}$, where $\tau = (s_0, a_0, r_0, \ldots, s_L, a_L, r_L)$, and $s_0 \in \rho_0$. The state-value function is the excepted return when an agent starts from a state s and follows the policy π, which is defined as

$$V^\pi(s) = \mathbb{E}_\pi[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s]. \tag{4.18}$$

To quantitatively estimate offline performance of DRL on $\mathcal{D}$, policy value v^π is a common metric [225, 226, 227] that computes the expected state value of initial states with the distribution ρ_0, which can be denoted as

$$v^{\pi}(\rho_0) = \mathbb{E}_{s_0 \sim \rho_0}[V^{\pi}(s_0)]. \tag{4.19}$$

In particular, we incorporate basal glucose control into $\mathcal{M}$ as follows.

States

In an AP system, BRs are frequently adjusted according to the real-time CGM measurements and the information of daily activities. Therefore, we model the states using the features extracted from a CGM sequence during the past hour and the external events recorded by insulin pumps or smartphone apps for diabetes management. Based on the feature processing in previous work [102, 251, 252, 212, 214, 253] and the results in model validation, we consider a total of seven features for glucose patterns, including the current BG level, the mean, maximum, and minimum values of the sequence, the maximum difference between adjacent measurements, and percentages of hypoglycaemia and hyperglycaemia. The cyclical encoding of timestamps, time and amount of last carbohydrate ingestion, and meal insulin bolus are also included in the states.

Actions

The action space is continuous, defined by the amount of BRs. In offline settings, we do not need to explicitly specify the range for random exploration.

Rewards

We design a reward function based on the clinical targets of TIR, TAR, and TBR, which are the standard metrics of glucose control recommended by the International Consensus [254]. In particular, TIR refers to the percentage of time that a T1D subject spends within the euglycaemic range (70-180 mg/dL). TBR is the percentage of time spent in the hypoglycaemic region and can be divided into level 1 (54-70 mg/dL) and level 2 (below 54 mg/dL). Similarly, TAR stands for the percentage of time spent in hyperglycaemia and

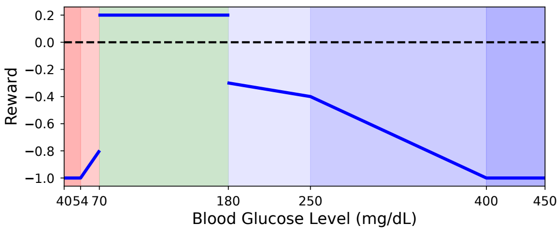


Figure 4.12: Reward function based on the clinical metrics of TAR, TIR, and TBR in glucose control.

can be divided into level 1 (180-250 mg/dL) and level 2 (above 250 mg/dL). Hence, we use a piecewise function with multiple slopes to compute a reward r_t using the current BG level at s_t and the intervals of clinical targets, as depicted in Figure 4.12, which penalises the agent when BG levels move toward hyperglycaemia or hypoglycaemia [251]. We denote this reward function as a TIR reward since the agent receives high rewards if good TIR is achieved. Another TBR reward function is used in OPE to estimate hypoglycaemia performance, which assigns -1 to a BG level below 70 mg/dL and 0 otherwise. We terminate episodes when a BG level is below 40 mg/dL or above 450 mg/dL, indicating that there is a medical emergency.

4.4.2 Offline Deep Reinforcement Learning

Double DQNs and DDPG have been proved to effectively improve glucose control in our previous work [251, 253, 252], as described in Section 4.2.2 and 4.3.2. The advantages of these two algorithms have been combined in recent studies on deep actor-critic methods for continuous control, which is known as the twin delayed deep deterministic policy gradient (TD3) [255]. By using target networks, delayed policy updates, and target policy smoothing, TD3 outperformed DDPG in benchmark environments [255]. Furthermore, combining a behaviour cloning regularization, TD3 has been demonstrated to be a minimalist method to achieve state-of-the-art performance in offline tasks [218], which has much lower complexity and computational cost when compared with other offline DRL algorithms [220, 221]. Therefore, we choose this variant of TD3 to learn personalised glucose control for T1D subjects using previously collected clinical data.

As an actor-critic approach, we first formulate the state-action value function (i.e., critic) Q^{π} updated by the Bellman equation as follows:

$$Q^{\pi}(s_t, a_t) = r_{t+1} + \gamma \mathbb{E}_{s_{t+1} \sim \mathcal{T}(s_{t+1}|s_t,a_t)}[V^{\pi}(s_{t+1})], \tag{4.20}$$

which is the expected return when an agent starts from a state s_t and takes action s_t. Parameterised by deep neural networks with parameters θ and ϕ, we denote the Q function and policy (i.e., actor) by Q_θ and π_ϕ, respectively. By doing this, we update Q_θ by TD learning with the temporarily frozen target Q-network $Q_{\theta'}$ and actor network $\pi_{\phi'}$. To reduce overestimation bias, TD3 uses twin Q functions (i.e., Q_{θ_1} and Q_{θ_2}) to select a less biased value estimate in each updating step. In offline settings, we sample a mini-batch $\mathcal{B}$ with M transitions $d = (s_t, a_t, s_{t+1}, r_{t+1})$ from $\mathcal{D}$ to calculate the loss $\mathcal{L}$ of TD learning as follows

$$\mathcal{L}(\theta_i) = \mathbb{E}_{d \sim \mathcal{B}}[(r_{t+1} + \gamma \min_{j=1,2} Q_{\theta'_j}(s_{t+1}, \pi_{\phi'}(s_{t+1}) + \epsilon) - Q_{\theta_i}(s_t, a_t))^2], \tag{4.21}$$

where $\epsilon \sim \texttt{clip}(\mathcal{N}(0, \sigma))$ is a clipped random noise to mitigate overfitting and smooth estimation in the deterministic policy. Aiming to reduce extrapolation error and distributional shift, a behaviour cloning term is introduced in the policy gradient of actor update, which can be denoted as follows

$$\mathcal{J}(\phi) = \mathbb{E}_{d \sim \mathcal{B}}[\lambda Q_{\theta_1}(s_t, \pi_\phi(s_t) - (a_t - \pi_\phi(s_t))^2], \tag{4.22}$$

where $\mathcal{J}$ is the loss function; $\lambda = \alpha(\mathbb{E}_{d \sim \mathcal{B}}|Q_{\theta_1}(s_t, a_t)|)^{-1}$ is a normalization factor calculated by the mean absolute values of Q_{θ_1}. The overall process of developing offline DRL is summarised in Algorithm 4.

4.4.3 Off-Policy Evaluation

The goal of OPE is to estimate the performance of DRL models using historical datasets and thus to rank and select polices, which enables us to assess the personalised glucose control algorithms without conducting actual clinical trials. Particularly, we choose FQE

Algorithm 4 Developing Offline DRL

```
Input: Randomly initialised $\theta_1$, $\theta_2$, and $\phi$, historical
training data $\mathcal{D}_{\text{train}}$, interval to delay policy update $t_d$
Output: Learned policy $\pi_\phi$
Set target networks: $\theta'_1 \leftarrow \theta_1$, $\theta'_2 \leftarrow \theta_2$, $\phi' \leftarrow \phi$
for steps $t \in 1, 2, \dots, T_{\text{DRL}}$ do
    Sample a mini-batch $\mathcal{B}$ from $\mathcal{D}_{\text{train}}$
    Update $Q_{\theta_i}$ for the critic using $\mathcal{L}(\theta_i)$ in Equation (4.21), for $i = 1, 2$
    if $t \mod t_d = 0$ then
        Update $\pi_\phi$ for the actor using $\mathcal{J}(\phi)$ in Equation (4.22)
        $\theta'_i \leftarrow (1 - \mu)\theta'_i + \mu\theta_i$, for $i = 1, 2$
        $\phi' \leftarrow (1 - \mu)\phi' + \mu\phi$
    end if
end for
```

as the OPE method with the implementation by Paine *et al.* [225], considering it provided accurate and robust estimation in healthcare settings [227]. Given a evaluation policy π with parameters ϕ, FQE initialises a critic with parameters ψ and retrains it using bootstrapping targets of the Bellman equation and a supervised learning loss as follows

$$\mathcal{L}(\psi) = \mathbb{E}_{d \sim \mathcal{B}}[(r_{t+1} + \gamma Q_{\psi'}(s_{t+1}, \pi_\phi(s_{t+1})) - Q_\psi(s_t, a_t))^2]. \tag{4.23}$$

The pseudo-code of OPE is presented in Algorithm 5, where the output is the estimated state-value function $\widehat{V}^\pi(s)$. Assuming a group of candidate policies $\{\pi_1, \pi_2, \dots, \pi_N\}$ with different hyperparameters is obtained in offline DRL by Algorithm 4, we can apply OPE to estimate a set of state-value functions $\{\widehat{V}^{\pi_1}(s), \widehat{V}^{\pi_2}(s), \dots, \widehat{V}^{\pi_N}(s)\}$. Using a validation set with the initial state distribution of ρ_0^{val}, we estimate policy values $\{\widehat{v}^{\pi_1}(\rho_0^{\text{val}}), \widehat{v}^{\pi_2}(\rho_0^{\text{val}}), \dots, \widehat{v}^{\pi_N}(\rho_0^{\text{val}})\}$ by Equation (4.19), as scalar scores to rank the candidate policies. Various reward functions can be used as input r' to evaluate model performance in different ways. As a result, the optimal policy $\pi*$ with the best scores is selected. Then, the control performance can be measured by either policy value $\widehat{v}^{\pi^*}(\rho_0^{\text{test}})$ for testing data or clinical metrics obtained in clinical or *in silico* trials. Figure 4.11 illustrates this process.

Algorithm 5 Developing OPE

```
Input: Randomly initialised $\psi$, historical training data
$\mathcal{D}_{\text{train}}$, update interval $t_u$, policy to be evaluated $\pi_\phi$, reward function $r'$
Output: Estimated $\widehat{V}^{\pi}(s) = Q_\psi(s, \pi(s))$
Set target networks: $\psi' \leftarrow \psi$
for steps $t \in 1, 2, \dots, T_{\text{OPE}}$ do
    Sample a mini-batch $\mathcal{B}$ from $\mathcal{D}_{\text{train}}$
    Update $Q_\psi$ for the critic using $\mathcal{L}(\psi)$, reward function $r'$ in Equation (4.23)
    if $t \mod t_u = 0$ then $\theta' \leftarrow \theta$ end if
end for
```

4.4.4 Offline Datasets

We conducted the experiments using two offline datasets to validate the clinical efficacy of the proposed algorithm, of which the details are summarised as follows. In both datasets, the BG levels were measured by CGM every five minutes.

In Silico Data

An *in silico* dataset was generated by the UVA/Padova T1D simulator (S2013) [81], which includes nine months of data of 10 virtual adults and 10 virtual adolescents. To emulate the variations of insulin sensitivity, we introduced a set of additional intra-subject variability [216] by adjusting meal intake protocols and the parameters of physiological models in the simulator. Specifically, the meal times of breakfast, lunch, and dinner followed normal distributions with mean values of 7:00, 14:00, and 21:00, respectively, and a standard deviation of 30 minutes. The corresponding carbohydrate amount for the three types of meals also followed normal distributions with mean values of 70, 110, and 90 grams, and a CV of 10%. The misestimation of carbohydrate counting was assumed to follow a uniform distribution with an interval of [0.7, 1.1]. The variability of insulin sensitivity, meal absorption, and carbohydrate bioavailability were respectively set to 30%, 10%, and 30%.

During data generation, the virtual subjects used personalised BR profiles and LGS [189] to control basal insulin and employed a SBC to compute meal insulin [228]. The LGS method has been demonstrated to significantly reduce the exposure to hypoglycaemia in clinical trials [256] and therefore is used as the baseline method in this case. We divided the *in silico* dataset into a development set containing the first six-month data and an unseen testing set containing the remaining three months of data to provide an unbiased evaluation. The last two months of data in the development set were used as a hold-out validation set, while the training set included the first four-month data.

Clinical Data

The publicly available OhioT1DM dataset [100] was employed to the analyze the proposed framework. It contains data of 12 people with T1D over an eight-week clinical trial. Each participant wore a Medtronic Enlite CGM sensor to measure BG levels and a Medtronic 530G or 630G insulin pump to deliver basal and bolus insulin, where a personalised BR profile was used. Notably, the T1D participants frequently set temporary BRs during the self-management to manually adjust basal insulin delivery, including zero BR for suspension, to meet insulin needs for a specified period of time, such as physical activities. Thus, we used such personalised BR (PBR) control as the baseline method, in order to reflect the performance of real-world glucose control.

4.4.5 Experimental Setup and Evaluation Metrics

The offline DRL algorithms developed by *in silico* data were evaluated by both simulation and OPE with the same meal scenarios and variability. Aiming to investigate the clinical performance, we first initialised the simulator using the initial state of the whole testing set for each subject and delivered basal insulin with the control strategy of the offline DRL algorithms through a three-month simulation. Secondly, we evaluated the offline DRL algorithms with OPE and investigated how well the OPE estimation matches actual policy values. In this case, we initialised the simulator using the initial state of each episode in a testing set (ρ_0^{test}) and obtained actual policy values by calculating rewards and state-values through *in silico* simulation for each episode.

To evaluate the outcomes of the three-month simulation, we use a group of clinical metrics. Besides the aforementioned TBR (level 1 & 2), TIR, and TAR (level 1 & 2), we employed LBGI and HBGI to indicate the risk of adverse glycaemic events, while CV is used to reflect glycaemic variability. The mean of BG levels measured by CGM is also presented. We performed the paired t-test to indicate the statistical significance after using the Shapiro-Wilk test to confirm normality.

For the real clinical data, the offline DRL algorithms were evaluated by OPE only since the clinical trials on the same T1D subjects cannot be performed. All the deep learning algorithms were developed by Python 3.8, PyTorch 1.9, and NVIDIA GTX 1080 Ti GPU. The episodes and transitions of datasets were structured by the d3rlpy framework [257]. The hyperparameters are listed in Table C.4 of Appendix C .

4.4.6 Performance on In Silico Dataset

Clinical Metrics

Table 4.5 and 4.6 respectively present the results (Mean±STD) of glucose control for the adult group and adolescent group through the three-month simulation, which was evaluated by the clinical metrics. It is worth noting that, when compared with the LGS baseline method, the offline DRL algorithm significantly enhanced TIR for the two virtual cohorts and achieved smaller TAR and TBR. In particular, level 1 TBR, level 1 TAR, and level 2 TBR decreased in both groups, while level 2 TAR was maintained in the adult group. Meanwhile, we observed that the offline DRL algorithm reduced the HBGI and LBGI, indicating a lower risk of hyperglycaemia and hypoglycaemia, and exhibited smaller mean CGM glucose and CV scores, indicating that BG concentrations are more stable. This comprehensive analysis suggests that the offline DRL algorithm effectively improved glucose control for the subjects in the *in silico* dataset.

OPE Quality

To investigate whether OPE is a reliable method for policy evaluation, we analyze the quality of OPE by a direct measure of rank correlation, which has been widely adopted in existing studies [225, 226, 227]. Figure 4.13 depicts the scatter plot of normalised OPE scores and actual policy values, where the TIR and TBR rewards were used for the adult

Table 4.5: Performance of glucose control on the virtual adult cohort through a three-month simulation

Metrics	LGS	Offline DRL
TIR (70 – 180 mg/dL) (%)	75.7 ± 6.1	$78.1 \pm 6.7^{\dagger}$
TAR (> 180 mg/dL) (%)	21.3 ± 6.4	$19.5 \pm 6.8^{*}$
Level 1 (181 – 250 mg/dL) (%)	19.5 ± 5.1	$17.4 \pm 5.4^{*}$
Level 2 (> 250 mg/dL) (%)	2.1 ± 2.0	2.1 ± 1.9
TBR (< 70 mg/dL) (%)	3.0 ± 1.5	$2.4 \pm 1.7^{*}$
Level 1 (54 – 69 mg/dL) (%)	1.9 ± 0.9	1.6 ± 1.1
Level 2 (< 54 mg/dL) (%)	1.1 ± 0.8	0.8 ± 0.5
LBGI	0.87 ± 0.41	0.76 ± 0.48
HBGI	4.24 ± 1.29	3.98 ± 1.67
CV (%)	31.0 ± 3.9	30.4 ± 3.0
Mean CGM glucose (mg/dL)	144 ± 10	141 ± 13

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

Table 4.6: Performance of glucose control on the virtual adolescent cohort through a three-month simulation

Metrics	LGS	Offline DRL
TIR (70 – 180 mg/dL) (%)	57.4 ± 12.7	$59.9 \pm 8.6^{*}$
TAR (> 180 mg/dL) (%)	38.4 ± 14.5	36.7 ± 9.8
Level 1 (181 – 250 mg/dL) (%)	24.5 ± 2.1	$23.8 \pm 2.6^{*}$
Level 2 (> 250 mg/dL) (%)	14.6 ± 14.4	12.2 ± 9.1
TBR (< 70 mg/dL) (%)	4.2 ± 2.6	$3.4 \pm 2.2^{\dagger}$
Level 1 (54 – 69 mg/dL) (%)	2.4 ± 1.2	$2.0 \pm 0.9^{\dagger}$
Level 2 (< 54 mg/dL) (%)	1.8 ± 1.8	$1.4 \pm 1.5^{\dagger}$
LBGI	1.51 ± 1.39	$1.24 \pm 1.10^{\dagger}$
HBGI	9.82 ± 6.69	8.37 ± 3.27
CV (%)	39.5 ± 4.1	38.1 ± 4.2
Mean CGM glucose (mg/dL)	171 ± 35	$165 \pm 18^{*}$

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

and adolescent cohorts. We performed the Spearman correlation analysis and obtained rank coefficients ρ of 0.98 ($p < 0.001$) and 0.91 ($p < 0.001$) for TIR and TBR rewards, respectively, indicating high correlation and good ranking statistics. The solid green line stands for the results of the linear regression between the two variables, and the shaded area is a 95% CI. A regression coefficient of 0.99 ($p < 0.001$) was achieved. These results demonstrate that the OPE method estimated accurate policy values and can be used to

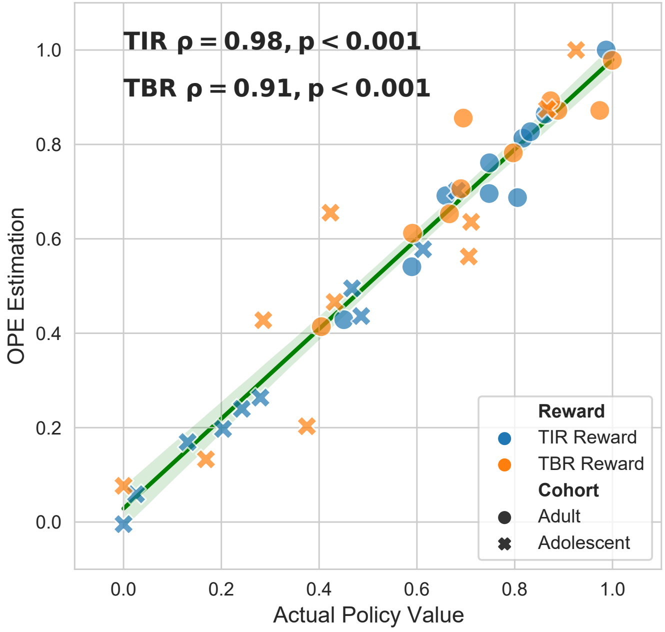
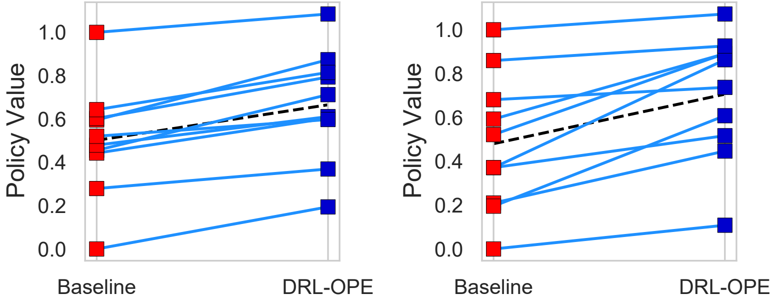


Figure 4.13: Scatter plot of the comparison between OPE estimation and actual policy values for the two cohorts and the two reward functions.

evaluate model performance in offline settings.

Policy Values

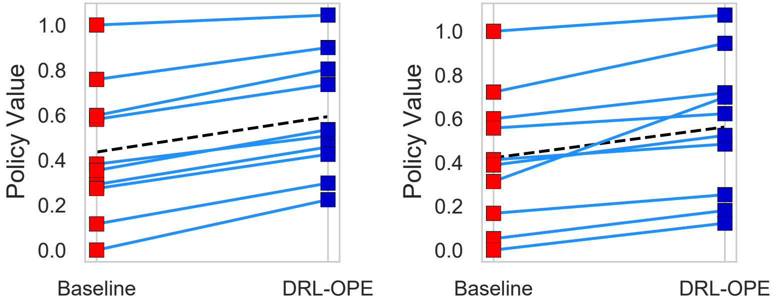
The ladder plots in Figure 4.14 and 4.15 show the improvement of the normalised policy values achieved by the proposed offline DRL algorithm in the adult and adolescent groups, respectively, when compared with the LGS baseline method. Of note, the offline DRL algorithm enhanced the policy value for each T1D individual in the two groups. Specifically, as indicated by the black dashed lines, the mean policy values significantly increased by 32.6% ($p < 0.01$) and 47.0% ($p < 0.01$) for the TIR and TBR rewards, respectively, in the adult group, while the mean policy value was significantly improved by 36.2% ($p < 0.01$) for the TIR reward and 33.5% ($p < 0.01$) for the TBR reward in the adolescent group. These results indicate that the offline DRL algorithm is expected to improve glucose control by simultaneously increasing TIR and reducing TBR for the virtual cohorts, which are consistent with the clinical results reported in Table 4.1 and 4.2. Figure 4.16 shows examples of BG trajectories and corresponding basal insulin delivered by the offline DRL



(a) TIR reward

(b) TBR reward

Figure 4.14: Comparison of policy values between the LGS baseline and the offline DRL algorithm for each T1D subject in the virtual adult group.



(a) TIR reward

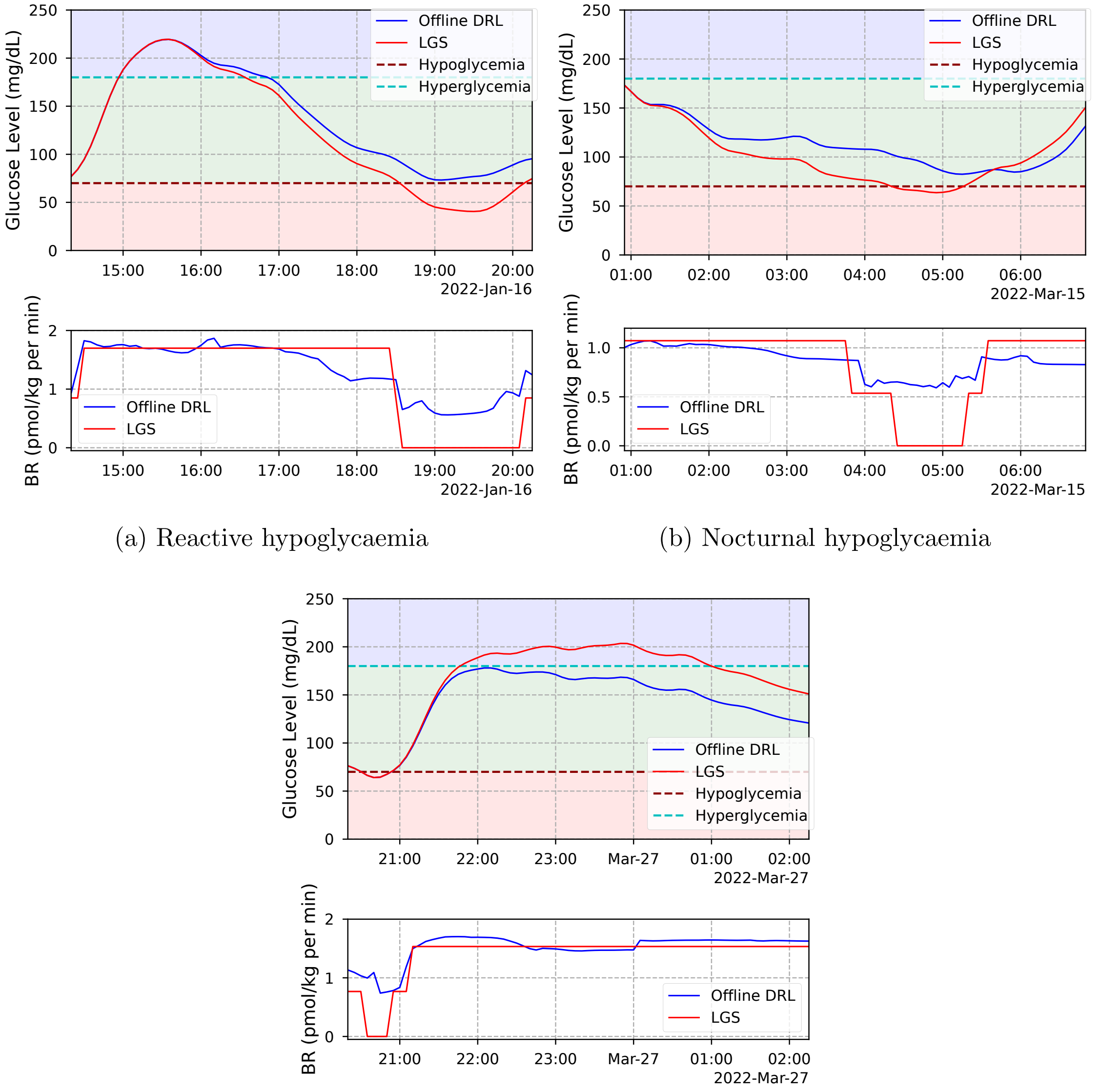
(b) TBR reward

Figure 4.15: Comparison of policy values between the LGS baseline and the offline DRL algorithm for each T1D subject in the virtual adolescent group.

and LGS algorithms. With the same initial state, the use of offline DRL algorithm successfully avoided severe reactive hypoglycaemia, nocturnal hypoglycaemia, and postprandial hyperglycaemia. The offline DRL control promptly adjusted BRs when the measured BG levels tended to move outside of the target range, resulting in higher policy values for both reward functions in the same episodes.

4.4.7 Performance on Clinical Dataset

Figure 4.17 depicts the normalised policy values of the PBR baseline method and offline DRL algorithm for real T1D subjects. A notable increase in policy values for each subject was observed in terms of the two reward functions. Compared with the PBR baseline, the



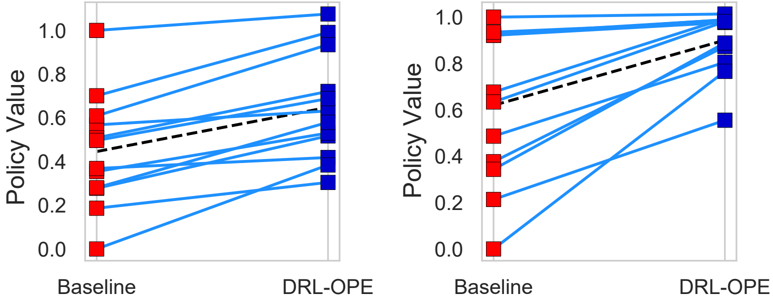
(a) Reactive hypoglycaemia

(b) Nocturnal hypoglycaemia

(c) Postprandial hyperglycaemia

Figure 4.16: Examples of BG trajectories and corresponding BRs controlled by the offline DRL algorithm and the LGS baseline method. The use of the offline DRL algorithm prevented the potential adverse glycaemic events that occurred in the LGS control.

offline DRL algorithm substantially improved the mean policy values by 45.3% ($p < 0.01$) for both the TIR and TBR rewards, as shown by the black dashed line. The TIR and TBR results (Mean±STD) of PBR on the OhioT1DM dataset are $63.5\% \pm 9.7$ and $3.3 \pm 2.3\%$, respectively. These values are close to the results of the virtual adult and adolescent groups (Table 4.1 and 4.2) and indicate that the *in silico* simulation reflect the real-world scenarios well. Therefore, we anticipate that using the trained offline DRL models can



(a) TIR reward (b) TBR reward

Figure 4.17: Comparison of policy values between the PBR baseline and the offline DRL algorithm for each T1D subject in the clinical dataset.

outperform the temporary BR settings of the PBR control and achieve better glucose control for the T1D subjects in the OhioT1DM dataset.

4.4.8 Discussion

To the best of our knowledge, this is the first study that applies offline DRL or OPE to glucose control in T1D management, and this is also the first study that combines offline DRL and OPE to solve a healthcare problem. Due to different datasets and various experimental settings, it is difficult to draw a head-to-head comparison of the numeric results of clinical metrics or policy values. However, in our previous work [251] (Section 4.2), we identified that optimising glucose control by adjusting a single hormone (i.e. basal insulin) is a challenging task. The reported TIR and TBR results on the *in silico* dataset (Table 4.5 and 4.6) are comparable with those by the online DRL algorithm after training agents through thousands of simulated days. Although a two-step TL framework was proposed in [251] to mitigate the high demand for personalised data, it still needs to fine-tune the policy with online interactions, where an undertrained agent may produce dangerous actions at the beginning of clinical trials. Moreover, the proposed DRL algorithm conservatively updated the policy by behaviour cloning regularization with a weighting factor α, where a smaller α means the update is more inclined to imitation learning. As shown in Figure 4.16, the BRs of the DRL algorithm were close to those of the LGS method, and notable differences were only observed when there was a risk of adverse glycaemic events.

Although this feature may lead to limited performance improvements, it prevents the DRL policy from deviating too far from behaviour polices π_β, aiming to avoid out-of-distribution actions and provide better safety guarantees for healthcare applications.

Several studies [102, 212, 213] added meal insulin bolus into action space and obtained new control strategies for total insulin delivery through long-term random exploration (e.g., millions of simulated hours), which is not feasible for real-world scenarios. In addition, the number of transitions of meal insulin bolus is quite limited (i.e., three or four times per day), while a wide range of carbohydrate content in food ingestion requires a large action space. The public use of dual-hormone AP for glucagon delivery has not been approved, and there is no existing dataset containing data of glucagon delivery. Hence, these hormones have not been considered in developing personalised offline DRL control yet, due to insufficient historical data. The scenarios considered in this study are the same as the real use cases of most AP systems [258]. That is, users manually enter carbohydrate amount into insulin pumps that calculate meal insulin bolus with bolus calculators, while basal insulin patterns can be automatically alternated by built-in personalised control algorithms.

Moreover, all the existing studies on DRL [102, 251, 252, 212, 214, 253] validated their models by simulators rather than actual clinical datasets, and their therapeutic benefits for real T1D subjects were unproven. Expert assessment [202] and a Bayesian framework [259] with Markov-Chain Monte Carlo strategy to estimate physiological models trace by trace (i.e., episode by episode) [260] were used to evaluate supervised learning algorithms on glucose control, but these assessment methods are either costly or time-consuming, especially when the size of testing data is large. In this regard, OPE provides a convenient method to evaluate control algorithms on clinical data and can be adopted by many existing models without much difficulty.

The model implementation is essential to the decision support systems for people living with T1D in daily self-management. Thus, aiming at on-device deep learning for model inference, we implemented the actor network of the offline DRL model on an iOS smartphone by converting the PyTorch model into the TensorFlow Lite format through the

Open Neural Network Exchange framework. Then we analyzed the run-time and memory usage of the embedded model using a customised diabetes management app (Figure 4.11), as described in Section 6.5.3. The experiment was repeated 50 times. After the app received a BG measurement from Dexcom G6 CGM, it took an average of 5.2 ms and 2.1 MB of memory to compute a BR for basal insulin delivery.

Such end-to-end implementation of a personalised control algorithm would incentive the development of the do-it-yourself artificial pancreas (DIY AP) [13, 261], in which people with T1D are able to build AP systems by themselves. However, the existing controllers in DIY AP, such as OpenAPS, AndroidAPS, and Loop, adjust BRs based on fixed physiological parameters and simple formulas but lack personalised algorithms to meet real-world challenges of inter- and intra-subject variability [262]. By employing the proposed offline DRL framework, the users can train, update and evaluate personalised insulin control algorithms using their own historical data collected during daily self-care.

4.5 Conclusion

With the aim of overcoming the challenge of blood glucose control in T1D, we first proposed a novel DRL algorithm for optimising basal insulin and glucagon delivery in AP systems. DRNNs are applied to the structure of double DQNs to develop personalised models through a two-step framework that involves TL. When compared with LGS, the proposed methodology significantly improves glycaemic outcomes in a virtual adult and adolescent population. This works shows that the proposed approach has the potential to be adopted in a clinical setting.

To benefit T1D subjects on MDI or SAP therapy as well, we proposed a novel meal insulin bolus advisor that uses the actor-critic DDPG architecture with multiple DNNs, which was trained by the two-step learning framework and prioritised memory replay. When compared with the standard therapy of insulin bolus calculation, the experimental results of the *in silico* trials indicate the promising performance of the DRL model, which significantly improved the TIR and reduced the risk of hypoglycaemia and hyperglycaemia for virtual cohorts of people with T1D.

Employing a safe offline process to build and implement personalised control models, we proposed a novel offline DRL framework to optimise BRs for basal insulin delivery. A TD3 model with behaviour cloning regularization and an FQE-based OPE method were adopted to develop the control algorithms, which ensure that the model training, validation, and testing can be completed offline without performing actual clinical trials or requiring the assessment by human experts. The proposed algorithm was validated on *in silico* and clinical datasets, which significantly enhanced TIR while reducing TBR and TAR and improved other clinical targets in the three-month simulation. A promising increase in policy values was also noted in the OPE analysis for both datasets.

We envision that the proposed algorithms would benefit T1D subjects on MDI, SAP, and AP systems and have huge potential to be adopted in a clinical settings of glycaemic control.

Chapter 5

Generating Personalised Glucose Data with GANs

Time series data generated by CGM sensors offer unparalleled opportunities for developing data-driven approaches, especially deep learning-based models, in diabetes management. Although these approaches have achieved state-of-the-art performance in various fields such as glucose prediction in T1D (Chapter 3), challenges remain in the acquirement of large-scale individual data for personalised model training due to the elevated cost of clinical trials and data privacy regulations. In this chapter, we introduce GluGAN, a framework based on GANs to generate personalised glucose time series with conditional inputs that consist of self-reported events. Employing RNN modules, the proposed framework uses a combination of unsupervised and supervised training to learn temporal dynamics in latent spaces. Aiming to assess the quality of synthetic data, we apply post-hoc RNNs to compute discriminative and predictive scores. In the experiments, GluGAN achieved better discriminative and predictive scores when compared with four baseline GAN models. The efficacy of data augmentation is evaluated by the use case of BG prediction with three classic machine learning-based predictors. Using the training sets augmented by GluGAN significantly reduced RMSE and MAE for the predictors over 30 and 60-minute horizons. This work provides GluGAN, an effective method to generate synthetic glucose time series that are highly similar to real CGM data. GluGAN has the potential to be used for evaluating the effectiveness of automated insulin delivery algorithms and as a digital twin to substitute for pre-clinical trials. The work in this chapter has led to the following journal article:

- **T. Zhu**, K. Li, P. Herrero and P. Georgiou, "GluGAN: Generating Personalized Glucose Time Series Using Generative Adversarial Networks," Submitted to *IEEE Journal of Biomedical and Health Informatics.*

5.1 Introduction

The application of deep learning, or other data-driven algorithms, in diabetes management is often challenging, since training these models usually requires collecting large-scale datasets in months-long clinical trials, a problem which is referred to as the cold start issue (Chapter 2). In this regard, glucose-insulin simulators, such as the OHSU T1D simulator [263] and the UVA/Padova T1D simulator [81], have been proposed to enable cost-effective *in silico* trials, which allow generating synthetic glucose time series using predefined virtual cohorts. Although the UVA/Padova T1D simulator has been demonstrated to be representative of a T1D population in a clinical trial [211], a major limitation of current glucose simulators is the lack of personalisation for T1D individuals, due to the large intra- and inter-subject variability [264, 19]. Visentin *et al.* used a Bayesian method to fit the UVA/Padova T1D simulator to one-day data collected from inpatient clinical trials with specific admission patterns, where each participant received three meals per day with the same meal-times and carbohydrate amounts [265]. However, exploiting outpatient data of T1D subjects under free-living conditions to develop a simulator for personalised data generation [265] is still an open problem. Moreover, personalised physiological simulation enables the development of digital twins [266, 267] for people with T1D, which has the potential to be used to assess therapeutic efficacy, predict glycaemic outcomes of intervention treatment, and reduce the need for actual clinical trials.

To this end, GAN-based frameworks can provide effective solutions. In recent works, GAN-based frameworks for sequential data generation have attracted significant attention. In [268], a GAN model was first applied to generate music using bidirectional RNNs with LSTM cells. Esteban *et al.* [269] extended LSTM-based GANs to the recurrent conditional GAN (RCGAN), which allow generating real-valued medical time series data with conditional inputs and a differentially private training procedure. Based on dilated convolutional neural networks, WaveGAN was proposed to produce synthetic audio [270]. Yoon *et al.* [271] proposed TimeGAN by introducing an embedding network and a recovery network in the GAN architecture to learn hidden temporal dynamics. The performance was tested in multiple time series datasets, including sinusoidal sequences, stock prices, energy data, and discrete events. In a more recent study, TimeGAN was applied to generate synthetic hypoglycaemic events to tackle the issue of imbalanced data in glycaemic classification [114]. Previously, we also explored a modified GAN-based model to extract feature maps from a multivariate input and forecast glucose levels [151].

In this chapter, we propose GluGAN, which to the best of our knowledge, is the first GAN framework that allows generating realistic glucose time series, and represents a first step towards data-driven personalised T1D simulators. Different from existing simulators based on fixed glucose profiles, GluGAN can learn glucose patterns for T1D individuals and generate personalised glucose data based on outpatient T1D datasets. The conditional inputs of GluGAN are daily entries manually recorded in T1D management, including SMBG measurements, carbohydrates from meal intake, and insulin delivery. The GAN model uses a combination of losses to learn temporal patterns of time series data. Three clinical datasets are employed to evaluate the quality of synthetic data and their usefulness in data augmentation. However, the interpretability and causality of the proposed method are still under-researched, which are essential factors for estimating the efficacy of glucose control algorithms in pre-clinical studies. Therefore, the current application of this work is a practical data augmentation tool for machine learning-based glucose prediction algorithms, instead of a fully-functional T1D simulator.

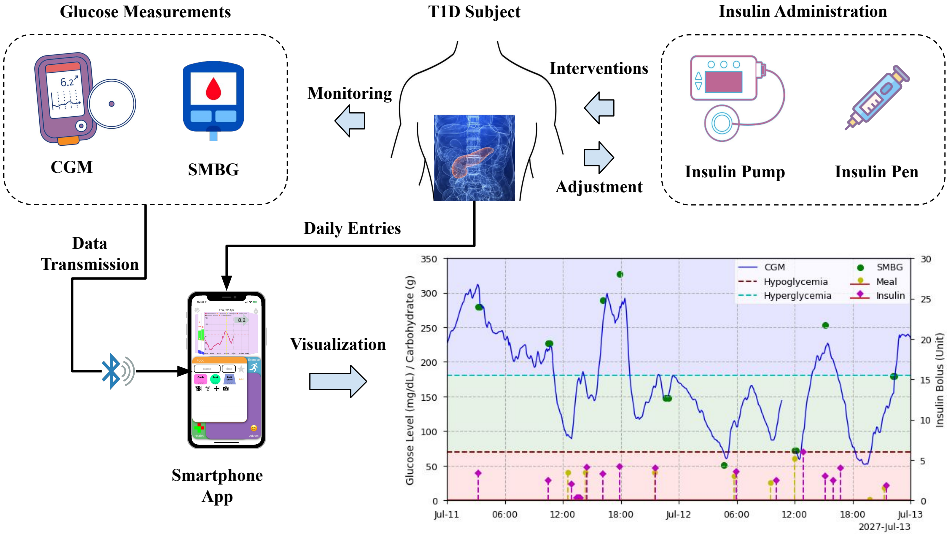


Figure 5.1: Illustration of a T1D management system. A smartphone app is used to communicate with the glucose monitoring devices via Bluetooth connectivity, collect daily logs, and visualise historical profiles and current trends of glucose levels. The right bottom plot shows two-day multivariate time series data of a clinical T1D subject in the OhioT1DM dataset.

5.2 Problem Formulation

Given personalised data from a clinical T1D dataset, we first slice the multivariate time series by a sliding window and obtain input sequences of length L. We assume an input sequence starts from a time-step ts and ends with a time-step t, where $ts = t + 1 - L$; and a vector at t is the current input of a RNN.

As shown in Figure 5.1, meal ingestion and insulin delivery can cause significant glucose fluctuations. Furthermore, CGM is usually calibrated with SMBG measurements when sensor replacement occurs. Discrepancies between CGM and SMBG are usually observed, especially in the postprandial period, which is a common phenomenon called meal-related glucose differences [272]. Meal, insulin, and SMBG data are therefore highly correlated to glucose dynamics and can provide rich information for generating realistic glucose time series. Hence, we convert these features into continuous conditional inputs $\mathbf{C}$ and attach them to CGM time series $\mathbf{G}$. Let $\mathbf{x}$ denote the model input of real data. We have $\mathbf{x}_{ts:t} = [\mathbf{G}_{ts:t}; \mathbf{C}_{ts:t}] \in \mathcal{R}^{4 \times L}$. Similarly, the model input of synthetic data is the concatenation of a random vector $\mathbf{Z}^{sp}$ sampled from a stochastic process and conditional inputs, which is denoted as $\mathbf{z}_{ts:t} = [\mathbf{Z}^{sp}_{1:L}; \mathbf{C}_{ts:t}] \in \mathcal{R}^{4\times L}$.

In this context, GluGAN aims at learning a density similar to the distribution of ground-truth data, when conditioned on the same auxiliary information. This is the objective of a standard GAN framework, which can be defined as follows [84]:

$$\min_{\hat{p}} \mathrm{JS}(p(\mathbf{G}_{ts:t}|\mathbf{C}_{ts:t})||\hat{p}(\mathbf{G}_{ts:t}|\mathbf{C}_{ts:t})) \tag{5.1}$$

where JS is the Jensen–Shannon divergence to measure of similarity between two probability distributions; p stands for the density function of the distribution over real data, while $\hat{p}$ is an approximate distribution of the generator's outputs. Jensen–Shannon divergence is a symmetric version of conventional Kullback–Leibler divergence, which is recommended to derive the adversarial loss of GANs [273]. However, the Jensen–Shannon divergence requires an optimal value for discriminator, i.e, perfect adversary, which is difficult to be obtained in unsupervised learning.

Fortunately, temporal relationships of time series data can guide the generation of sequential data, especially for the time series with high correlation across timesteps. Therefore, we introduce another objective function that focuses on the step-wise conditional distributions, which can be formulated as [271]:

$$\min_{\hat{p}} \mathrm{KL}(p(G_t|\mathbf{G}_{ts:t-1}, \mathbf{C}_{ts:t-1})||\hat{p}(G_t|\mathbf{G}_{ts:t-1}, \mathbf{C}_{ts:t-1})) \tag{5.2}$$

where KL denotes the Kullback–Leibler divergence. Although this divergence is an asymmetric measure, it can be optimised by supervised learning with maximum-likelihood estimation [271]. Therefore, a supervised loss is employed in the adversarial training to learn the transition dynamics.

5.3 GluGAN Architecture

To generate realistic personalised glucose time series, we develop GluGAN by modifying the stand GAN architecture in three ways. First, we use the aforementioned loss for supervised learning to optimise the objective in Equation (5.2) and combine it with the unsupervised adversarial loss to optimise the objective in Equation (5.1). In this case, the model not only learns to generate data with similar distribution but also to capture the temporal dynamics of time series. Secondly, in addition to the generator discriminator, we introduce three other modules into GluGAN, including an embedding network, a recovery network, and a supervisor network. The embedding and recovery networks are used for auto-encoding to project time series data into a lower-dimensional latent space, where both adversarial learning and supervised learning are performed, aiming to improve generation performance for high-dimensional time series [271]. The supervisor network is used to learn the step-wise dynamics with the targets derived from real sequences. These three additional modules are jointly trained with the generator and discriminator. Each module consists of a four-layer RNN with GRU cells, of which the hyperparameters are determined in model validation. Finally, we employ three conditional input features [269] to indicate the underlying glycaemic states, including carbohydrate amount of meal ingestion, bolus insulin, and SMBG measurements.

Figure 5.2 depicts the overall architecture of the proposed GluGAN. For the auto-encoding purposes, we employ the embedding network to convert the real input features to latent representations $\mathbf{h}$ and reconstruct glucose data $\widetilde{G}$ through the recovery network $\mathcal{R}$. Similarly, given a random input vector, the generator $\mathcal{G}$ outputs the synthetic embedding vector $\hat{\mathbf{e}}$, which is disciplined by the supervisor $\mathcal{S}$ to learn step-wise temporal dynamics for the synthetic latent vector $\hat{\mathbf{h}}$. Then, we obtain the synthetic glucose time $\hat{G}$ by the mapping function of the recovery network. Instead of directly comparing the outcomes of the generator with real data, the discriminator $\mathcal{D}$ of GluGAN performs classification in the latent space. We denote the outputs of the discriminator by $y_t, \hat{y}_t \in \{0, 1\}$ for the real and synthetic model input data, respectively.

Correspondingly, a total of three losses is used to optimise the weights of GluGAN.

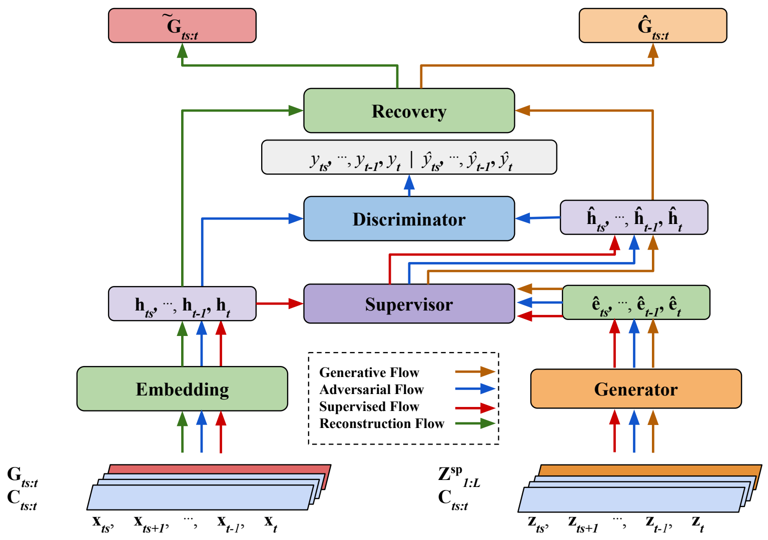

$\tilde{G}_{ts:t}$
$\hat{G}_{ts:t}$
Recovery
$y_{ts}, \cdots, y_{t-1}, y_t \mid \hat{y}_{ts}, \cdots, \hat{y}_{t-1}, \hat{y}_t$
Discriminator
$\hat{h}_{ts}, \cdots, \hat{h}_{t-1}, \hat{h}_t$
$h_{ts}, \cdots, h_{t-1}, h_t$
Supervisor
$\hat{e}_{ts}, \cdots, \hat{e}_{t-1}, \hat{e}_t$
Embedding
Generative Flow
Adversarial Flow
Supervised Flow
Reconstruction Flow
Generator
$G_{ts:t}$
$C_{ts:t}$
$x_{ts}, x_{ts+1}, \cdots, x_{t-1}, x_t$
$Z^{sp}_{1:L}$
$C_{ts:t}$
$z_{ts}, z_{ts+1}, \cdots, z_{t-1}, z_t$

Figure 5.2: System architecture of the proposed GluGAN. The data flows corresponding to glucose time series generation, adversarial training, supervised learning, and reconstruction are marked with orange, blue, red, and green arrows, respectively.

To obtain reliable conversion between latent space and glucose features, a reconstruction loss $\mathcal{L}_R$ is applied to train embedding and recovery modules, which is given by:

$$\mathcal{L}_R = \mathbb{E}_{\mathbf{x}\sim p}[\sum_t \left\|\widetilde{G}_t - G_t\right\|_2]. \tag{5.3}$$

Meanwhile, supervised learning with a loss $\mathcal{L}_S$ is performed to minimise the divergence in Equation (5.2). This loss aims at minimising the step-wise differences between synthetic and real latent vectors, when the current synthetic embedding vector $\hat{\mathbf{e}}_t$ are conditioned on the real latent sequences at previous timesteps ($\mathbf{h}_{ts:t-1}$), which is formulated as:

$$\mathcal{L}_S = \mathbb{E}_{\mathbf{x}\sim p,\mathbf{z}\sim\hat{p}} \sum_t \left\|\mathbf{h}_t - \mathcal{S}(\hat{\mathbf{e}}_t, \mathbf{h}_{ts:t-1})\right\|_2, \tag{5.4}$$

where $\mathcal{S}$ represents the function of the supervisor network. Similar to the standard GAN framework, the discriminator is trained to be better at discriminating real from synthetic data, while the generator is designed to generate sequences that are indistinguishable from real glucose time series. Hence, we treat the optimization as a two-player minimax game. The unsupervised losses of adversarial training are calculated using the classification results $y_t, \hat{y}_t$, which are given by:

$$\begin{aligned} \mathcal{L}_U^{\mathcal{G}} &= \mathbb{E}_{\mathbf{z}\sim\hat{p}} \sum_t \log(1-\hat{y}_t), \\ \mathcal{L}_U^{\mathcal{D}} &= -\mathbb{E}_{\mathbf{x}\sim p} \sum_t \log y_t - \mathbb{E}_{\mathbf{z}\sim\hat{p}} \sum_t \log(1-\hat{y}_t), \end{aligned} \tag{5.5}$$

where $\mathcal{L}_U^{\mathcal{G}}$ and $\mathcal{L}_U^{\mathcal{D}}$ denote the unsupervised losses of the generator and discriminator, respectively.

5.4 Model Development

Algorithm 6 Developing GluGAN to Generate Glucose Time Series

```
Input: a preprocessed training set $\mathcal{D}_{tr}$ and synthetic inputs $\mathcal{P}_Z$; iteration numbers $\mathcal{T}_R, \mathcal{T}_S, \mathcal{T}_J$; loss ratios $\lambda_1, \lambda_2$; iterations of the inner loop $k$, threshold of discriminator loss $l_D$.
Model training
for iterations in $\mathcal{T}_R$ do
    Sample mini-batches from $\mathcal{D}_{tr}$
    Update the weights of the embedding and recovery networks by minimising loss $\mathcal{L}_R$
end for
for iterations in $\mathcal{T}_S$ do
    Sample mini-batches from $\mathcal{D}_{tr}$ and from $\mathcal{P}_z$
    Update the weights of the supervisor network by minimising $\mathcal{L}_S$
end for
for iterations in $\mathcal{T}_J$ do
    for iterations in $k$ do
        Sample mini-batches from $\mathcal{D}_{tr}$ and $\mathcal{P}_z$
        Update the weights of the supervisor and generator by minimising $\lambda_1\mathcal{L}_S + \mathcal{L}_U^{\mathcal{G}}$
        Update the weights of the embedding and recovery networks by minimising $\lambda_2\mathcal{L}_S + \mathcal{L}_R$
    end for
    Sample mini-batches from $\mathcal{D}_{tr}$ and $\mathcal{P}_z$
    if $\mathcal{L}_U^{\mathcal{D}} > l_D$ then
        Update the weights of the discriminator by minimising $\mathcal{L}_U^{\mathcal{D}}$
    end if
end for
Model testing
Given the batch of $\{\mathbf{z}_{ts:t}\}$ from a testing set $\mathcal{D}_{te}$
Obtain synthetic glucose time series by $\hat{\mathbf{G}}_{ts:t} = \mathcal{R}(\mathcal{S}(\mathcal{G}(\mathbf{z}_{ts:t})))$
```

Data preproccessing is an essential step to obtain high-quality multivariate time series from clinical datasets. We first remove outliers for each input feature with a set of maximum and minimum thresholds based on physiological features. In particular, we exclude negative values for each feature, insulin doses above 50 units, and glucose values above 500 mg/dL. Notably, there are many missing gaps in CGM measurements (Figure 5.1), due to various reasons, such as sensor replacement and calibration, sensor noise, and signal loss. Thus, we perform linear interpolation to fill the gaps in the middle of CGM sequences and use linear extrapolation for missing data samples on the tails [101, 128, 151]. We also exclude the CGM sequences with a gap longer than 15 minutes. All the input features are normalised to a range of [0,1] with the min-max normalization.

Algorithm 6 presents the details of the model development. Training the GluGAN model (Figure 5.2) requires optimising all the network modules with the loss functions defined in Equations (5.3), (5.4), and (5.5). Given the numbers $\mathcal{T}_R, \mathcal{T}_S, \mathcal{T}_J$ of training iterations for embedding learning, supervised learning, and joint learning, respectively, we optimise the embedding and recovery networks with ground truth data and $\mathcal{L}_R$, and then train the supervisor module alone with $\mathcal{L}_S$. Finally, all five modules are jointly trained with the combinations of unsupervised and supervised losses. Specifically, two hyperparameters λ_1, λ_2 are employed to adjust the ratios of $\mathcal{L}_S$ when combined with reconstruction and unsupervised losses. To achieve a Nash equilibrium for the two-player non-cooperative game and avoid the discriminator becoming too strong in adversarial training [274], we update the generator more frequently with an inner loop and update the discriminator only if $\mathcal{L}_U^{\mathcal{D}}$ is above a predefined threshold l_D [268]. During the testing phase, GluGAN can generate synthetic glucose data with the batch inputs of testing data. Table D.1 of Appendix D summarises the values of the hyperparameters used in this work.

The performance of GluGAN is tested on three real datasets: the OhioT1DM [100], ARISES, and ABC4D datasets, which are collected from a number of T1D subjects over months of trials with different clinical settings. The details of the datasets are described in Section 3.3.1. The demographic characteristics of the three datasets are shown in Table B.4 of Appendix B.

5.5 Experiment Setup

5.5.1 Data Splitting and Analysis

During model development, we split datasets into training sets $\mathcal{D}_{tr}$ and testing sets $\mathcal{D}_{te}$. The OhioT1DM dataset is provided with a training set and a testing set for each T1D subject, which respectively contain data of around 40 and 10 days [100]. As for the subjects in ARISES and ABC4D datasets, we use the first 80% data as training sets and the rest 20% data as hold-out testing sets. It is a common split method for developing machine learning algorithms in BG prediction, which guarantees that future information is not involved in current model inference [19]. For the training sets in each dataset, we use the first 75% data for model training, while the last 25% data are used as hold-out validation sets to tune hyperparameters. We generate synthetic datasets $\hat{\mathcal{D}}_{tr}$ and $\hat{\mathcal{D}}_{te}$ for training and testing sets, respectively, aiming at following the train-on-synthetic and test-on-real (TSTR) routine [269] to test model performance.

In Figure 5.3, we plot the autocorrelogram of glucose time series using consecutive sequences with a minimum length of three days . It is to be noted that, for the three considered datasets, the glucose data have high autocorrelation when time lags are smaller than 105 minutes. Therefore, it is important to introduce the autoregressive prior and supervised learning loss into GluGAN model to learn step-wise temporal dynamics of glucose time series.

5.5.2 Evaluation Metrics

To comprehensively evaluate the performance generative models, we use a set of commonly employed metrics in previous work on GAN frameworks and time series data [269, 271]. The similarity and diversity of synthetic and real testing glucose data (i.e., $\hat{\mathcal{D}}_{te}$ and $\mathcal{D}_{te}$) are qualitatively visualised by PCA [275] and t-distributed stochastic neighbor embedding (t-SNE) [276]. We convert temporal L-dimension sequential data into two dimensions and plot the distributions, where each dot stands for a glucose sequence.

Moreover, quantitative analysis to measure the similarity and diversity is also per-

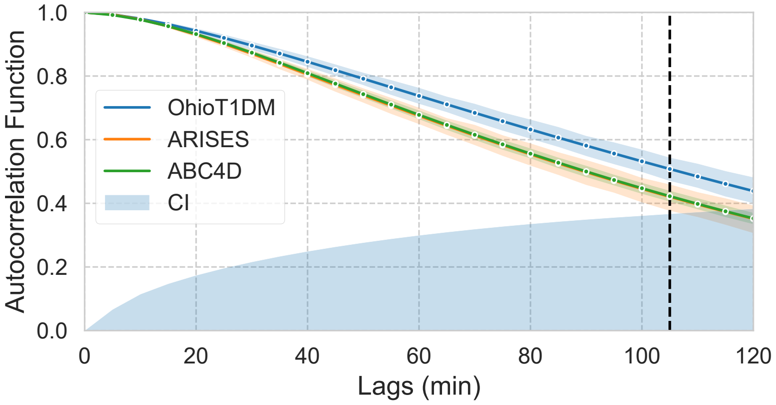


Figure 5.3: Autocorrelation function of glucose time series in the three datasets with lags up to 120 minuets. The autocorrelation function of the OhioT1DM, ARISES, and ABC4D datasets are respectively showed in blue, orange, and green solid lines with 95% bootstrap CIs. The blue shaded region indicates the 95% Bartlett CI, and any autocorrelation function outside this region is statistically different from zero. The vertical dashed line indicates the maximum time lag (105 min) with significant autocorrelation function for the three datasets.

formed. We use a post-hoc binary classifier, which is based on an RNN with two LSTM layers [271]. It is trained on both training sets (i.e., $\hat{\mathcal{D}}_{tr}$ and $\mathcal{D}_{tr}$) and tested on testing sets (i.e., $\hat{\mathcal{D}}_{te}$ and $\mathcal{D}_{te}$). Then we calculate a discriminative score as $|\text{Accuracy} - 0.5|$, where $\text{Accuracy} = (TP + TN)/(N + P)$; TP and TN are the number of true positives and true negatives, respectively; $(N + P)$ stand for all the outputs. A lower discriminative score indicates better performance of GAN models, which means that the classifier's accuracy is close to random guess, and synthetic data are indistinguishable from real data. Similarly, we introduce a step-wise predictive score with a post-hoc predictive model to predict next-step glucose value and calculate scores of RMSE. The predictive model is also constructed by an RNN model with two LSTM layers, which is trained on synthetic datasets ($\hat{\mathcal{D}}_{tr}$) and tested on real datasets ($\mathcal{D}_{te}$), i.e., TSTR. Low predictive scores indicate the synthetic data are useful in terms of prediction tasks. Besides using the RMSE scores, we also introduce a commonly used metric, MAE to evaluate the performance of each model in the experiments of glucose prediction with data augmentation.

5.5.3 Baseline Methods

We compare the performance of GluGAN, i.e., the quality of the synthetic data, with a group of GAN models with temporal settings in the literature. Specifically, the considered baseline methods, including TimeGAN [271], RCGAN [269], C-RNN-GAN [268], and WaveGAN [270]. Among these, TimeGAN, C-RNN-GAN, and WaveGAN use univariate glucose data, while GluGAN and RCGAN use multivariate inputs. We retain the main architectures of the baseline GAN models and tune the hyperparameters by the same hold-out validation datasets, according to the predictive scores. To avoid model overfitting, we apply early stopping to model training with a patience number of 50.

5.5.4 Model Training and Testing

We test the effect of GluGAN and data augmentation through a TL framework [128, 146]. Assuming only the first two-week glucose data are available in each training set, we combine these data as a global set to develop a population GluGAN model and fine-tune the whole model with individual training data of a hold-out subject. This particular length is selected because the lifespan of most commercial CGM sensors is within 14 days. Then the personalised GluGAN model generates two-week synthetic glucose data, which are combined with the original data to develop an augmented training set. We compare the performance of glucose prediction using augmented training sets (train-on-augmented and test-on-real (TATR)) and original training sets (train-on-real and test-on-real (TRTR)). Figure 5.4 depicts the TATR and TRTR process. Three predictors, including the post-hoc LSTM, DRNN [128], and SVR [160], are evaluated on the testing sets with 30 and 60-minute PHs. DRNN is a state-of-the-art model to accurately predict glucose levels for the OhioT1DM dataset in our previous work [128], which is based on three dilated RNN layers. We also apply the same TL framework to improve the performance of LSTM and DRNN predictors [128, 146] (Figure 5.4). SVR is a robust machine learning model in glucose prediction and is commonly used as a baseline method in the literature [19]. All the deep learning models are developed by TensorFlow 1.15 and Python 3.7, while SVR (radial basis function kernel) is deployed by scikit-learn 0.24. Training the deep neural

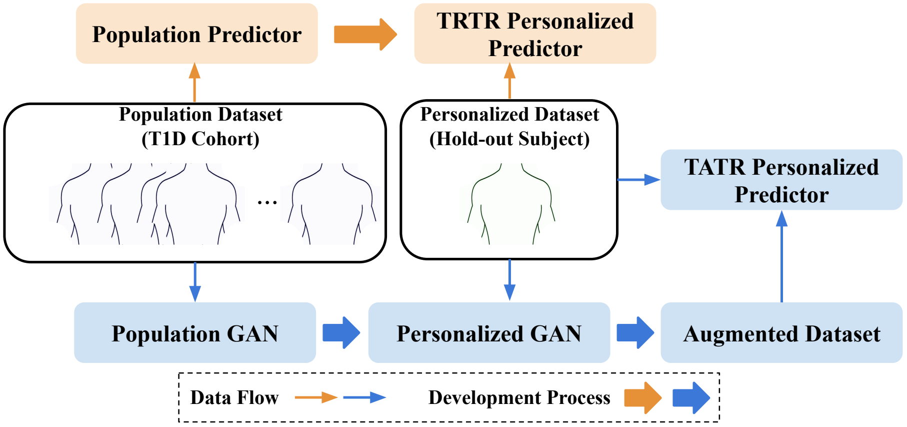

Population Predictor
TRTR Personalized Predictor
Population Dataset (T1D Cohort)
...
Personalized Dataset (Hold-out Subject)
TATR Personalized Predictor
Population GAN
Personalized GAN
Augmented Dataset
Data Flow
Development Process

Figure 5.4: Diagram of the TRTR (orange arrows) and TATR (blue arrows) in data augmentation experiments for glucose prediction. The TL framework is applied to develop personalised GluGAN and deep learning-based predictors (i.e., LSTM and DRNN). The thin and thick arrows indicate data flow and development process, respectively.

networks is accelerated by NVIDIA GTX 1080 Ti GPU.

5.6 Results and Discussion

Figure 5.5 shows the results of PCA and t-SNE analysis for three T1D subjects from OhioT1DM, ARISES, and ABC4D testing sets, respectively. It is to be noted that the distributions of synthetic and real glucose time series are highly overlapped, indicating good similarity.

In order to quantitatively evaluate the quality of synthetic data, we compared the performance of GluGAN with four existing GAN frameworks. We computed the statistical significance (p-value) by paired t-test after confirming the normality of distributions by the Shapiro-Wilk test. Figure 5.6a depicts the performance of discriminative scores for GluGAN and the considered baseline methods evaluated on the three clinical datasets. Notably, GluGAN achieved the smallest mean discriminative scores across all the datasets (0.17 ± 0.09, 0.16 ± 0.07, 0.13 ± 0.05; overall result: 0.15 ± 0.07) and obtained significant improvements, except for the OhioT1DM dataset, when compared with the baseline methods.

Similarly, Figure 5.6b shows the results of predictive scores. Compared with the

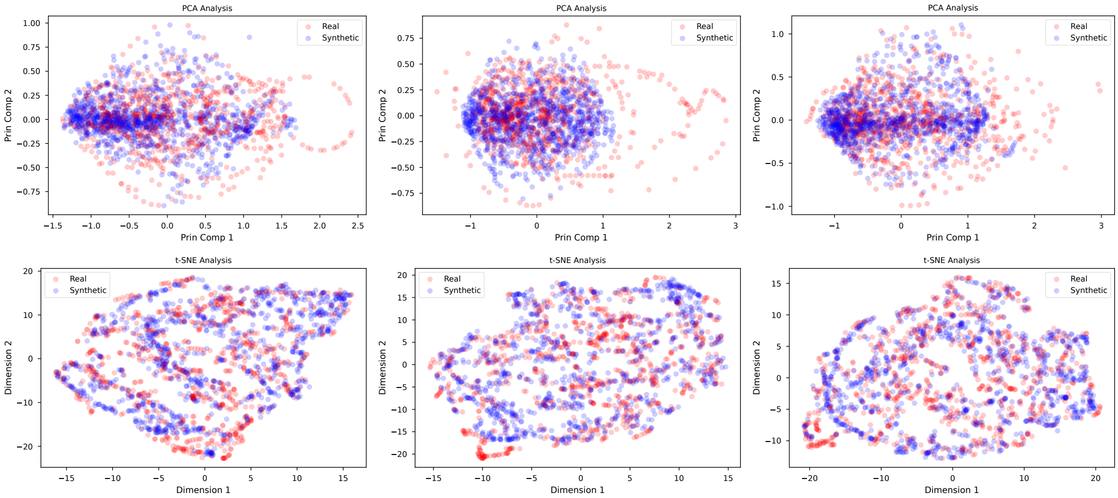


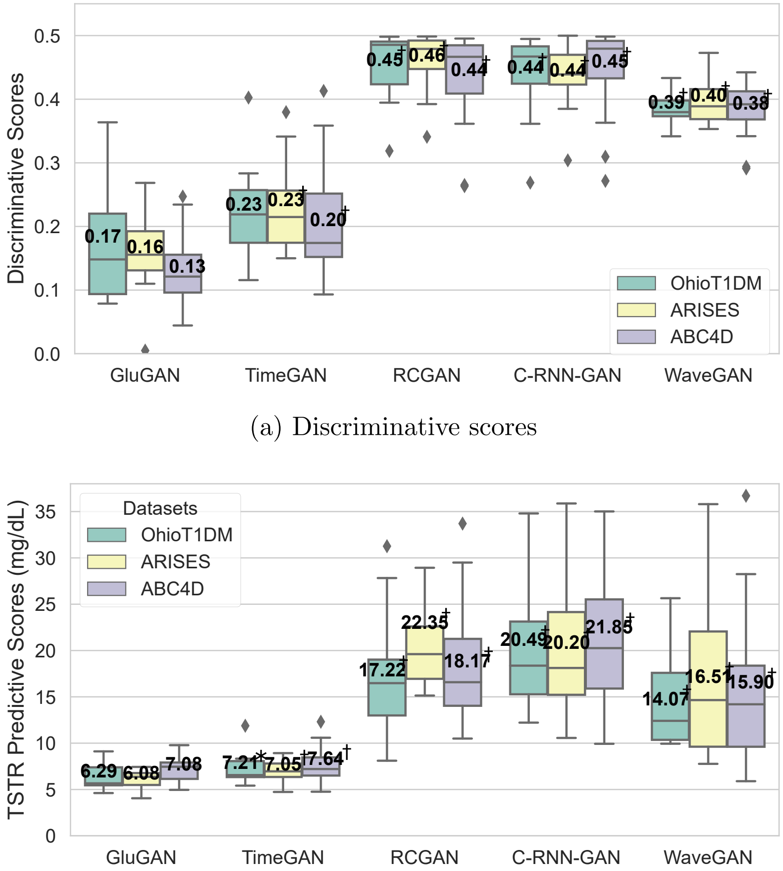
Figure 5.5: PCA (the top row) and t-SNE (the bottom row) analysis on the distributions of real and synthetic glucose sequences that are displayed in the red and blue dots, respectively. The columns from left to right show the plots for the OhioT1DM, ARISES, and ABC4D datasets, respectively.

considered baseline methods, GluGAN also significantly improved the performance and achieved better mean predictive scores of 6.29 ± 1.31, 6.08 ± 0.98, and 7.08 ± 1.26 mg/dL for the OhioT1DM, ARISES, and ABC4D datasets (overall result: 6.64 ± 1.30 mg/dL), respectively.

Figure 5.7 shows an example of a one-day period of real and synthetic glucose time series. It is observed that the synthetic curve passes through three of the four SMBG measurements and has trends and peaks that are highly correlated with the actual CGM measurements. In the experiments, we noted that excluding SMBG features did not have a significant impact on predictive scores but caused larger bias and degraded the overall mean discriminative scores by 0.02.

Table 5.1 presents the performance of glucose prediction with three data-driven prediction algorithms over 30 and 60-minute PHs. Each predictor is tested by TATR and TRTR routines. It is worth noting that the use of augmented training sets in TATR significantly reduced RMSE and MAE scores in each dataset.

To the best of our knowledge, this work is the first attempt to generate realistic T1D glucose time series based on a specifically designed GAN framework, i.e., GluGAN.



(a) Discriminative scores

(b) TSTR predictive scores

Figure 5.6: Performance of GAN models evaluated on the OhioT1DM, ARISES, and ABC4D datasets. **(a)**: Discriminative scores. **(b)**: TSTR predictive scores. The central lines of the boxplots indicate the median values, and the whiskers indicate the distance of 1.5 times the interquartile range. The mean values are shown in the center of the boxplots. Statistical significance is indicated as * for $p \leq 0.05$ and † for $p \leq 0.01$.

To prove the validity and evaluate the performance of the approach, three clinical datasets were employed. The visualization in Figure 5.5 and results in Figure 5.6a demonstrate that GluGAN is able to generate high-fidelity synthetic glucose data, of which the distributions are similar to those of real data. The results in Figure 5.6b indicate that the synthetic data preserved good temporal dynamics and can be useful in terms of glucose prediction tasks. It is to be noted that, by taking advantage of the autoregressive formulation and latent space auto-encoding, GluGAN and TimeGAN achieved much smaller discriminative and predictive scores than the other three baseline GAN models. We noticed that it is

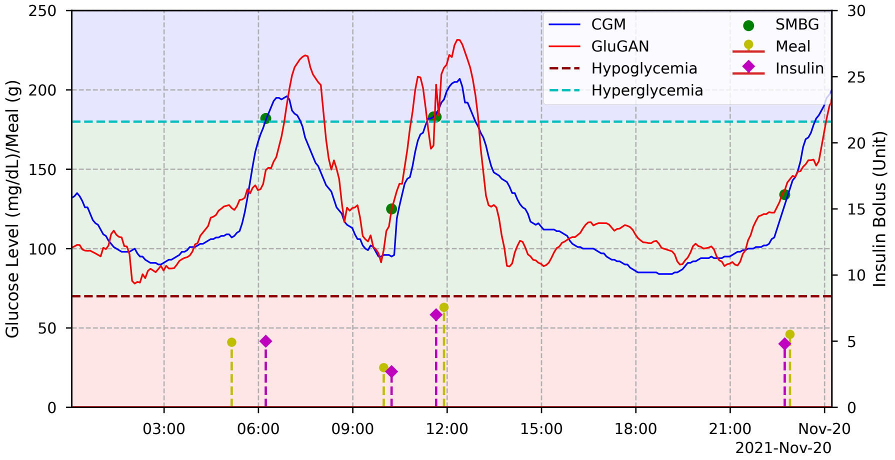


Figure 5.7: Visualization of synthetic glucose time series for a T1D subject in the OhioT1DM dataset over a day. The real CGM readings and the synthetic glucose values generated by GluGAN are displayed in blue and red solid lines, respectively. The hypoglycaemic, euglycaemic, and hyperglycaemic regions are marked by light red, green, and blue shaded areas, respectively. The conditional inputs of SMBG, carbohydrate of meal intake, and insulin bolus are shown in green dots, vertical yellow lines, and vertical magenta lines, respectively.

challenging to train the GAN model to generate glucose time series without supervised loss, mainly due to the high correlation cross timesteps (Figure 5.3). C-RNN-GAN used a feature-matching approach [274] with a supervised loss in generator training, aiming to match the hidden representations between real and synthetic data [268]. We also used this basic supervised loss in another baseline method, RCGAN; otherwise, the model would fail to generate realistic data. The authors of WaveGAN used the Wasserstein distance and a gradient penalty [277] to improve the loss of the generator [270]. Furthermore, to investigate whether GluGAN simply memorised training data and reproduced them during the generative phase, we conducted an analysis of maximum mean discrepancy with the radial basis function kernel [269]. We computed the maximum mean discrepancy scores for synthetic training data ($\hat{\mathcal{D}}_{tr}$) and synthetic testing data ($\hat{\mathcal{D}}_{te}$) and employed the Kolmogorov-Smirnov test with the null hypothesis that these two groups of scores are sampled from the same distribution. In the experiments, we have $p \gg 0.05$ for all the datasets, indicating the performance on the training data is not significantly better than that on the testing data.

Table 5.1: Performance of glucose prediction by LSTM, DRNN, and SVR trained on augmented (TATR) and original (TRTR) training sets. Statistical significance is indicated as * for $p \leq 0.05$ and † for $p \leq 0.01$.

Datasets	**Method**	**LSTM**		**DRNN**		**SVR**	
				PH = 30			
		RMSE	MAE	RMSE	MAE	RMSE	MAE
OhioT1DM	TATR	20.28 ± 2.42	14.42 ± 1.42	19.80 ± 2.35	14.10 ± 1.55	22.00 ± 5.26	15.01 ± 2.17
	TRTR	21.30 ± 2.43†	15.55 ± 1.57†	20.39 ± 2.48†	14.64 ± 1.71†	23.89 ± 4.51†	17.28 ± 2.19†
ARISES	TATR	21.57 ± 4.04	15.78 ± 2.88	21.02 ± 3.73	15.35 ± 2.63	24.61 ± 6.10	17.11 ± 3.76
	TRTR	22.81 ± 4.11†	16.86 ± 2.97†	21.69 ± 4.03	16.01 ± 2.98*	26.39 ± 7.38†	19.09 ± 5.01†
ABC4D	TATR	21.03 ± 2.47	14.42 ± 1.42	20.81 ± 2.45	15.07 ± 1.82	21.63 ± 2.15	15.59 ± 1.63
	TRTR	22.08 ± 2.45†	15.55 ± 1.57†	21.21 ± 2.52†	15.31 ± 1.86*	23.52 ± 2.24†	17.87 ± 1.95†
Datasets	**Method**			PH= 60			
		RMSE	MAE	RMSE	MAE	RMSE	MAE
OhioT1DM	TATR	33.80 ± 3.64	25.05 ± 2.96	33.43 ± 3.56	24.80 ± 2.72	35.47 ± 5.99	25.59 ± 3.70
	TRTR	35.61 ± 4.50*	26.64 ± 3.61*	34.38 ± 3.73*	25.37 ± 2.97	36.46 ± 5.87*	27.06 ± 3.66†
ARISES	TATR	37.10 ± 7.37	27.75 ± 5.41	36.54 ± 7.31	27.41 ± 5.44	39.10 ± 7.42	28.66 ± 5.62
	TRTR	38.98 ± 8.34*	29.28 ± 6.49*	38.19 ± 8.09	28.64 ± 6.22	39.23 ± 7.77	29.31 ± 6.17*
ABC4D	TATR	35.26 ± 4.89	26.37 ± 3.62	35.07 ± 4.85	26.28 ± 3.63	36.22 ± 4.76	26.74 ± 3.45
	TRTR	36.37 ± 4.95†	27.46 ± 3.74†	36.03 ± 4.90†	27.05 ± 3.62†	36.64 ± 4.50	27.91 ± 3.43†

We also applied GluGAN to increase the amount of available training data, and thus to enhance the performance of a glucose prediction algorithm. Particularly, we explored two deep learning predictors and a machine learning predictor with 30 and 60-minute PHs. As shown in Table 5.1, the use of augmented training sets reduced RMSE and MAE for all the predictors over the two PHs. In this case, GluGAN is an effective and model-agnostic solution to meet the challenge of limited personal data and the cold start issue for the development of data-driven models. To perform a fairer comparison in data augmentation, future work includes introducing two baseline methods to increase training data. One is to randomly oversample the training data, and the other is to add random noise to test whether the improvement in RMSE is because the models fit the random noise. Figure 5.7 shows the visualization of synthetic glucose over 24 hours, which was obtained by retraining the GluGAN model with an input length of one day. It is worth noting that the trends and peaks of the synthetic data are similar to that of the real CGM measurements, which may offer an estimation of ambulatory glucose profiles for T1D subjects with the SMBG regimen only.

5.7 Conclusion

In this chapter, we proposed a novel framework, GluGAN, based on GAN architectures and deep learning technologies for the generation of realistic blood glucose time series in T1D. In the reported experiments, we demonstrated that GluGAN is able to generate high-quality synthetic data and outperformed all the considered baseline GAN models, in terms of predictive and discriminate scores derived by post-hoc RNNs.

As an application of the proposed approach, we used GluGAN to increase the size of training data as a data augmentation technique for blood glucose prediction. In particular, we enhanced the prediction accuracy of different machine learning-based glucose predictors (LSTM, DRNN, and SVR) over 30 and 60-minute PHs.

The promising results of this work have demonstrated the feasibility of using GluGAN to improve data-driven decision support systems in T1D management. Finally, in combination with physiological models and clinical constraints, GluGAN has the potential to be employed as a personalised T1D simulator or a digital twin in future work.

Chapter 6

An IoMT Framework in Diabetes Management

The recent development of CGM and deep learning technology have been demonstrated to achieve the state of the art in BG prediction and glycaemic control, as described in Chapter 3 and 4. However, it is challenging to implement such algorithms in actual clinical settings to provide persistent decision support due to the high demand for computational resources, while smartphone-based implementations are limited by short battery life and require users to carry the device. In this chapter, we propose an IoMT-enabled wearable device that comprises a low-cost and low-power system on a chip (SoC) to perform Bluetooth connectivity and edge computing for decision support. In addition, we develop a smartphone app according to T1D user feedback in the ARISES trial (Section 3.5.1) to visualise BG trajectories and predictions, and desktop and cloud platforms to backup data and fine-tune models.

Considering BG prediction is essential to diabetes management, we build an embedded deep learning model using the FCNN framework proposed in Section 3.2 and implement it on the wearable device for real-time BG prediction and predictive hypoglycaemia detection. The embedded model achieved superior performance of RMSE, MAE, and gRMSE, and obtained the best accuracy for hypoglycaemia detection when compared with a group of machine learning baseline methods. Moreover, we performed hardware-in-the-loop *in silico* trials with 10 virtual T1D adults to test the whole IoMT system with predictive low-glucose management, which significantly reduced hypoglycaemia and improved BG control. This work provides an IoMT system for diabetes management, including a smartphone app, a desktop platform, a cloud platform, and a customised wearable wristband that provides wireless connectivity and edge computing for real-time decision support. The work in this chapter has led to the following journal articles:

- **T. Zhu**, L. Kuang, J. Daniels, P. Herrero, K. Li, P. Georgiou, "IoMT-enabled real-time blood glucose prediction with deep learning and edge computing," *IEEE Internet of Things Journal*, 2022.
- **T. Zhu**, C. Uduku, K. Li, P. Herrero, N. Oliver, P. Georgiou, " Enhancing self-management in type 1 diabetes with wearables and deep learning", *npj Digital Medicine*, vol. 5, no. 1, p. 78, 2022.

6.1 Introduction

With the rapid development of Internet of things, recent advances in CGM have been shown to enhance the treatment for people with T1D [34]. A CGM system comprises an implanted sensor to measure interstitial BG levels and a transmitter to send measurements to a receiver, such as a customised hardware box, smartphone, or smart watch with a fixed frequency (e.g., every five minutes). As a well-established paradigm of IoMT [278], CGM can also be combined with insulin pumps as SAP or AP. In this context, BG prediction can be used in closed-loop AP systems with model predictive control [139] and enables PLGM systems that have been proved to be effective for reducing hypoglycaemia in clinical settings [134].

Deep learning-based models have recently achieved superior performance in BG prediction [19]. Of note, by employing the latest deep learning technology, the increasingly complex models rely on a huge number of parameters, neurons, and layers for model inference. Thus, how to implement these models in actual clinical settings to bring actual therapeutic benefits is under-researched, which can be problematic since on-device inference with a large number of model parameters requires intensive computational resources and memory consumption.

The existing methods to implement deep learning models for BG prediction are mainly based on customised smartphone apps [96, 101, 158]. However, several limitations exist in these methods including lack of wearability, battery constraints, and the dependency on mobile operating systems. It is inconvenient for T1D users to carry smartphones or other handheld devices all the time, especially during high-intensity activities that would reduce the awareness of subsequent hypoglycaemia in T1D [279]. In addition, the battery level of smartphones and smartwatches are significantly drained because the prediction algorithms continuously run in the background with Bluetooth connectivity [280]. As a result, the decision support system will be unavailable when the devices run out of power. Moreover, the smartphone implementation is highly dependent on mobile operating systems, such as Android and iOS, and deep learning libraries, such as PyTorch [158] and TensorFlow Lite [96, 101]. Many existing apps for diabetes management suffer from the frequent updates of mobile operating systems. T1D users, especially the elderly population, may need to purchase extra expensive smartphones if the implementation does not support their own devices. Cloud implementation could be a solution to this problem, but it is largely limited by Internet connectivity since there are many daily scenarios suffering from poor coverage of WiFi and mobile signals. Thus, a power-efficient and low-cost wearable device based on edge computing [281, 282, 283] is preferred in T1D management to provide real-time BG prediction and predictive hypoglycaemia detection. The outcomes of this study also indicate the possibility of embedding deep learning algorithms into CGM devices (e.g., wearable transmitters).

6.2 IoMT Framework Overview

Figure 6.1 depicts an overview of the proposed system architecture in T1D management. There are three subsystems: 1) monitoring and decision support, 2) medical interventions, 3) platforms and servers, which are described in the subsequent sections. The IoMT-enabled wearable device is in the center of the monitoring and decision support system.

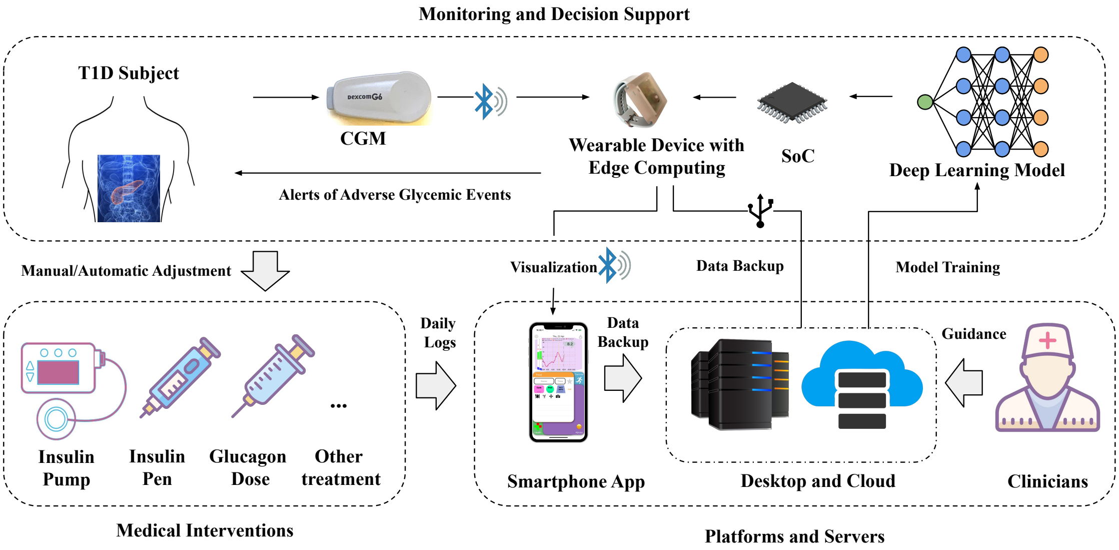


Figure 6.1: System architecture of the T1D management system with the proposed wearable device, which contains three subsystems as follows: monitoring and decision support, medical interventions, and platforms and servers. The wearable device is a part of the monitoring and decision support subsystem that can provide real-time measurement, BG prediction, and hypoglycaemia warning.

Communicating with the CGM via Bluetooth connectivity, the wearable device empowers a T1D user with real-time BG prediction and hypoglycaemia detection. Then the user can interact with the subsystem of medical interventions to adjust treatment. The data transmission between the wearable device and the platforms and servers aims at data visualization, data backup, and updating the embedded deep learning model.

Monitoring and Decision Support

As the core component of the proposed system, it contains a CGM sensor that measures BG levels every five minutes and transmits the real-time measurements to a specifically designed wearable wristband via Bluetooth low energy (BLE). The SoC of the wearable device performs the embedded deep learning algorithm to predict BG levels and detect forthcoming hypoglycaemic events. The historical CGM measurements and DNN weights are stored in the Flash memory, which can be accessed and updated by the platforms and servers. This essential subsystem can run solely without interactions with other devices to guarantee persistent and reliable decision support throughout day and night. In addition, thanks to a power-efficient design of SoC, the battery life of the wearable device (six months) is longer than that of the CGM sensor (10 days) and transmitter (three months).

Medical Interventions

Automatic control with the same SoC that enables Bluetooth communication with insulin pumps has been validated in our previous work [56]. In this chapter, we consider manual control to fit different clinical scenarios since insulin pumps are not widely used by people with T1D. Once receiving predictions and warnings from the wearable device, a T1D subject is allowed to seek necessary interventions in advance and manually adjust existing medical treatment.

Platforms and Servers

A smartphone app can connect with the wristband through Bluetooth to visualise current CGM reading, predictions, and historical BG trajectories, while recording daily activities, such as meals, excise, and health conditions. A desktop platform with a specifically designed graphical user interface (GUI) (Figure E.1 of Appendix E) is employed to train the deep learning models and backup collected data. It communicates with the wristband through USB ports and can upload data to the Amazon cloud storage, i.e, a bucket of Amazon S3. T1D users are allowed to perform these operations by themselves or with the guidance of healthcare providers or clinicians if needed. To facilitate users without a programming background, we deploy the deep learning models in the cloud using Amazon SageMaker. Thus, the models can be automatically trained with newly uploaded data on the cloud platform and downloaded from the cloud storage to the wearable device.

6.3 Problem Formulation and Feature Engineering

Denoting a BG level measured by CGM at timestep t as G_t, the target of prediction is to estimate a future BG value of G_{t+p}, where p is a PH (e.g., 30 minutes) normalised by resolution of CGM. To extract hidden representations, the input data contains a sequence of retrospective data $\mathbf{X_t}$ with a length of L, i.e., $\mathbf{X}_t = [\mathbf{x}_t, \mathbf{x}_{t-1}, \ldots, \mathbf{x}_{t-\Delta}] \in \mathbb{R}^{d \times L}$, where d is the dimension of the input features; $\mathbf{x}_t \in \mathbb{R}^{d \times 1}$ denotes the input vector at the timestep t; and $\Delta = L - 1$. Considering edge computing that delivers computation close to data sources, we derive all the input features from CGM measurements and corresponding timestamps in the monitoring and decision support system (Figure 6.1). The timestamps for a 24-hour period are converted into two types of time index to map seasonal patterns: min-max normalization with a range of $[0, 1]$ [128] and sine-cosine encoding [146]. The BG change over the PH is used as the learning target y_t to reduce underlying bias [101, 128], i.e., $y_t = f_n(G_{t+p} - G_t)$, where f_n is the min-max normalization to scale each feature.

Combining CGM sequences with time index, we perform feature selection during the validation phase. The best validation performance was obtained with the CGM time series $\mathbf{G}_t$ and min-max normalised timestamps $\mathbf{S}_t$, i.e., $\mathbf{X}_t = f_N([\mathbf{G}_t; \mathbf{S}_t])$. However, we notice that there is a large number of missing gaps in the historical CGM measurements, due to some inevitable reasons (e.g., sensor calibration and signal loss), which account for around 10% of the total length. Thus, we interpolate the missing CGM data in the middle of input sequences and extrapolates the missing CGM data at the tail to avoid involving future information in current predictions.

6.4 BG Prediction by Edge Evidential Neural Network

A challenge of implementing such models in actual clinical settings is the lack of evaluating the uncertainty and confidence of predictions. It is essential to determine whether a prediction is reliable and confident when a deep learning model aims to provide crucial decision support in a healthcare system. To this end, we build an embedded edge evidential neural network (E3NN) model to compute the lower bounds of each prediction, based on the FCNN framework proposed in Chapter 3. Figure 6.2 shows the architecture of the proposed deep learning model, consisting of a base model with a stack of RNN layers, an attention layer, a dropout layer, a dense layer, and an evidential output layer. The input of E3NN is a multivariate time series with CGM and timestamps, while the output comprises the parameters of the evidential distribution to compute prediction values and

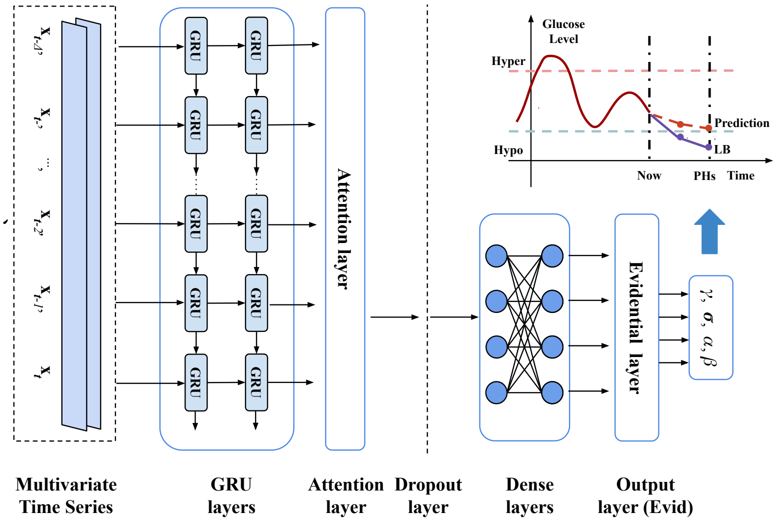


Figure 6.2: Block diagram of the proposed E3NN. The model input is a multivariate time series. The output of E3NN includes the four parameters $p[\gamma, \sigma, \alpha, \beta]$ of the posterior distribution to compute BG predictions with corresponding lower bounds.

lower bounds.

6.5 Model Implementation and customised Devices

6.5.1 Edge Computing

Compared with model implementation on the cloud, edge computing can offer more reliable real-time services on the wearable device with extremely low latency of decision making, which are not limited by Internet connectivity. Deep learning with edge inference is emerging research in the fast-growing areas of AI and Internet of things. Existing inference frameworks, such as TensorFlow Lite Micro [65] and CMSIS-NN [284], currently support a limited subset of operations and DNN layers. Therefore, we convert the E3NN TensorFlow models to C models based on the CMSIS-DSP library that offers high-performance APIs for math functions, such as matrix operations, and the firmware development is based on the latest nRF5 SDK v17.0.2.

The BLE SoC is based on an ARM Cortex-M4 Core with a tight memory budget (512 KB Flash and 64 KB SRAM). However, the SoC not only communicates with the front-end CGM transmitter through the BLE protocol but also runs the trained RNN model on the edge. Thus, we optimise the SRAM usage of the model inference. In particular, assuming the input of the embedded model involves L CGM readouts associated with timestamps, the first GRU layer processes a two-dimensional data sample at each timestep and repeats for L rounds. Each round of operations is dependent on the output from the previous iteration and cannot be computed in parallel for acceleration. However, as the second GRU layer runs in the same way except for using the output of the first GRU layer as input data, the operations of these two GRU layers can be pipelined. Thus, the cells of the stacked RNN layers at the same timestep are performed in one round and iterated L times. This interleaving process reduces the SRAM utilization since only one output needs to be temporally stored instead of an output vector with a length of L. The dropout layer only applies in the training phase, which is disabled in model inference and thus not implemented on the SoC.

We import RNN weights as 4-byte hex data as a raw format representing 32-bit floating-point numbers, aiming to maintain the prediction accuracy with less loss of precision compared with post-training quantization. These weights are fixed and stored in the Flash memory, which can be claimed as static constant variables.

6.5.2 Embedded System and Wearable Device Design

To meet the requirements raised for edge computation, the proposed system involves a lightweight and compact hardware design for a low-power and low-cost wearable device. It is embedded with four main peripherals including the LEDs, button and buzzer for essential user interactions as shown in Figure 6.3, where a Nordic SoC (nRF52832) is employed as the system controller. The entire system can be divided into two parts. The first part is a hardware power-gating circuit that includes a timer and a load switch to control the on/off state of the system, while the second part inside the power-gated region aims to save energy during the idle period.

Once a prediction is made and an adverse BG event is detected, the user can be notified through either the light or sound, generated by the LED and buzzer, whereas a

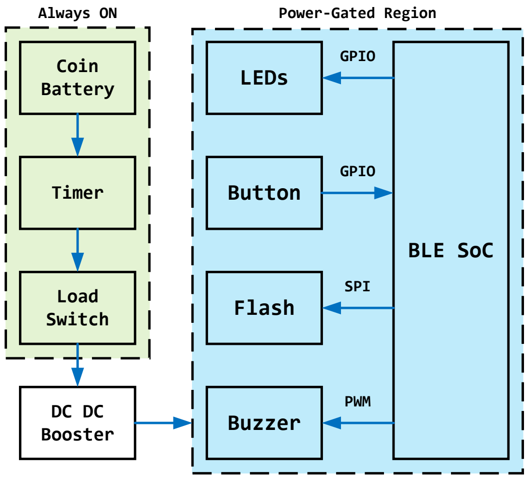


Figure 6.3: Block diagram of the proposed IoMT system in T1D management, which is powered by a coin battery and embedded with LEDs, Buttons, and a Buzzer for user interactions as well as a NOR Flash for data storage. The system employs a hardware power-gating circuit including a timer and a load switch for energy saving, which can maintain ultra-low-power during the idle period.

simple click on the button can stop the notification. In addition, to backup the CGM readouts for post-processing, a NOR Flash is employed that provides 16 MB memory capacity. Considering that each data sample transmitted by CGM every five minutes only contains 16 bytes, such Flash memory space can support long-term data storage for more than one year. The only drawback is that the Flash-type memory does not support random access, thus the writable address must be determined at the start point. To solve this, a binary search algorithm is implemented on the SoC which significantly improves the efficiency compared with searching exhaustively.

The BLE SoC can enter a soft power-down mode for energy saving before starting the next CGM readout. However, its peripherals can still consume some power if they are connected to the main supply. As a result of this, the power gating technique is applied in the system to shut off the current to the BLE SoC and its peripherals during the idle period. This is realised through a timer integrated circuit which generates a periodic power-gated signal to control a load switch. In addition, due to the need for user notifications, such a process may take a different amount of time. Because of this, an extra signal driven by the BLE SoC is connected to the timer integrated circuit to enter

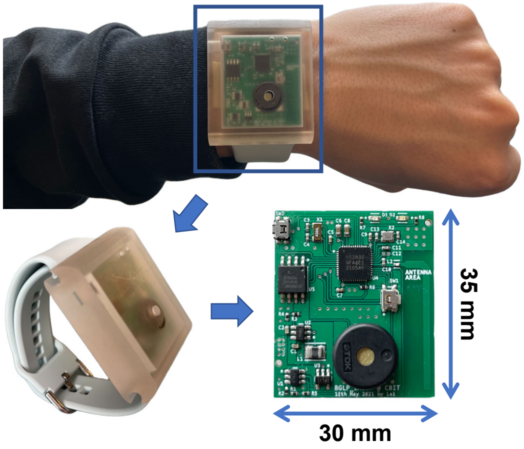


Figure 6.4: IoMT-enabled wearable device consists of a printed circuit board designed in a dimension of 35 mm x 30 mm and a transparent case manufactured by 3D printing[1].

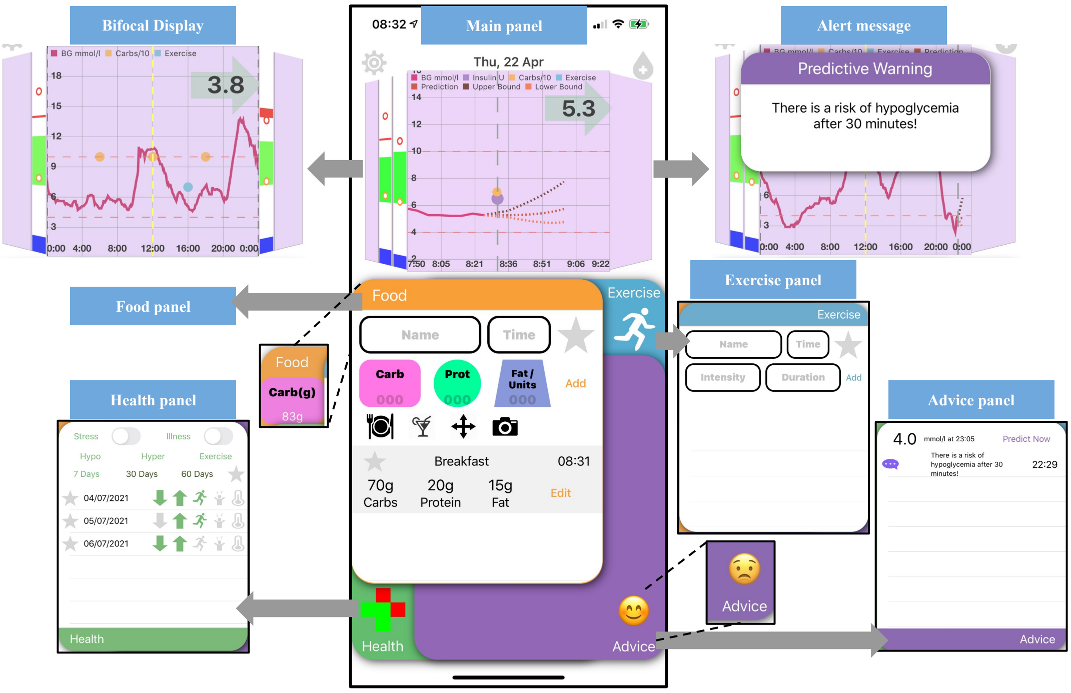
the shutdown mode.

Due to the compactness of the proposed system, the finial hardware is populated onto a 2-layer printed circuit board in a dimension of 35 mm x 30 mm as shown in Figure 6.4. The printed circuit board is inside a 3D printed case with a transparent appearance. The button for users to confirm hypoglycaemic events is located at the top-left edge. The black cylinder that occupies a large area of the printed circuit board is a buzzer. The size of the wearable device is close to a smartwatch (e.g., Apple Watch). It can be powered by a single coin battery (CR2302) with a lifespan of six months.

6.5.3 Smartphone App Design

In this chapter, a smartphone app is employed as a visualisation tool. As shown in Figure 6.5, the GUI is based on the ARISES app developed in our previous clinical study (Section 3.5.1). The ARISES app is based on the iOS operating system and integrates with Dexcom CGM (G5 or G6) and Empatica E4 wristband, which was designed by a multidisciplinary team consisting of diabetes clinicians and engineering specialists. In addition, the feedback of end users (i.e., the T1D participants) was collected in a series of focus groups, which specified the usability requirements and provided suggestions for

[1] The wearable device was designed and implemented by Mr. Lei Kuang.



There is a risk of hypoglycemia after 30 minutes!

Breakfast		08:31
70g Carbs	20g Protein	15g Fat

Edit

Figure 6.5: Graphical user interface of the smartphone app, where bifocal display [285] is used to present glucose trajectories[2]. The users are allowed to record daily activities through the food, exercise and health panels. The embedded deep learning algorithm predicts glucose levels and detects hypo- and hyperglycaemia with lower and upper bounds. Then the app sends corresponding warnings to users by notifications and haptic vibration and displays information in the advice panel. The corner icons of the food and advice panels can show the details of carbohydrate and the status of unread alert messages, respectively, when these panels are stacked at the bottom.

improving the GUI.

An common way to implement deep learning models is by smartphone app. Therefore, we implemented the ARISES model (Section 3.6.2) and analysed the performance of the app on an iPhone XS Max over 50 runs. The whole app has an initial storage size of 39.9 MB and consumed an average of 50.5 MB and 39.3 MB memory while running in foreground and background, respectively. The trained deep learning models were converted to mobile compatible format via TensorFlow Lite, which has a storage size of 1.2 MB. When the app received a new CGM measurement, it took 5.7 ms and 1.8 MB memory to compute real-time glucose prediction through model inference on the edge, which require one-hour historical data of CGM measurements, sensor wristband measurements,

[2] The design of the GUI was proposed by Prof. Robert Spence, and the diary modules were initially implemented by Mr. Ryan Armiger.

and daily entries (if any). Model fine-tuning is performed by Amazon S3 buckets and SageMaker in the Amazon Web Services cloud and requires at least one-week historical data.

6.6 Experiment Setup

We developed and evaluated the algorithms using three datasets collected from a number of T1D subjects in clinical trials, which are described in Section 3.3.1. The OhioT1DM dataset contains the training set and testing set of each T1D subject [100], which account for the data of around 40 days and 10 days, respectively. Similarly, each of the ABC4D and the ARISES datasets was divided into a training set that includes the first 80% data and a testing set with the last 20% data. For each training set, the last 25% data was used as a validation set for hyperparameter tuning. This setup can avoid introducing temporal dependencies into training and testing sets, which was commonly used in previous work [19]. The selected values of the hyperparameters are listed in Table E.1 of Appendix E.

We developed a personalised model for each T1D subject with 30-minute and 60-minute PHs and compared the proposed E3NN model against a group of baseline methods in the literature. The ARIMA [161] and SVR [160] were selected as two classic machine learning baselines [19], while the temporal convolutional network (TCN) [101], CRNN [96], LSTM [107], and Bi-LSTM [286], were employed as deep learning baselines. All the considered models were implemented by Python 3.8 and used the same input features, except for ARIMA that used CGM input only. We respectively applied statsmodels 0.12 and scikit-learn 0.23 libraries to build the ARIMA and SVR models. The deep learning models were developed by TensorFlow 2.2 and Keras 2.3. We trained them using an Adam optimiser and early stopping to mitigate overfitting, which was accelerated by NVIDIA GTX 1080 Ti GPU. Notably, to evaluate the performance of model inference on the wearable SoC, we sequentially fed input data to the embedded E3NN models through a universal asynchronous receiver-transmitter (UART) with the general serial port data transmission protocol.

We evaluated the accuracy of BG prediction using three classic metrics: RMSE, MAE, and gRMSE in mg/dL, which are defined in Section 3.3.3. According to the international consensus [287], a hypoglycaemic event is defined as three consecutive CGM measurements below 70 mg/dL. MCC is used to evaluate hypoglycaemia detection. It is a preferred metric in binary classifications since high MCC scores can be obtained only if the classifier performs well in all the categories of confusion matrix [288], which is denoted in Section 3.3.3.

6.7 Results and Discussion

6.7.1 Prediction Performance

BG Level Prediction

Table 6.1, 6.2, and 6.3 respectively present the results of BG level prediction for the OhioT1DM, the ABC4D and the ARISES datasets over 30-minute and 60-minute PHs. To indicate the statistical significance with respect to the considered baselines, we confirmed the normality of data distribution with Shapiro–Wilk test and employed paired t-test to compute p values. It is worth noting that the E3NN achieved the best RMSE, MAE, and gRMSE for all three datasets and obtained significant improvement, compared with the considered baseline methods. Particularly, the improvement of the E3NN methods on the OhioT1DM dataset is more significant than that on the other two datasets, which is possibly due to the high quality of the dataset with the smallest portion of missing CGM samples. We observe that the RMSE for the 60-minute PH is much higher than that for the 30-minute PH, because external events, such as meal intake and exercise, and internal changes in a T1D subject are more likely to occur within a longer period, which would have an impact on glucose dynamics.

Overall, the deep learning methods performed better than the classic machine learning baselines, except for CRNN. The RNN-based models, including LSTM and Bi-LSTM, exhibited better performance than TCN and CRNN that use CNN layers for feature extraction, where the LSTM performed best among the baseline model. In addition, it is

Table 6.1: Performance of the prediction models evaluated on the OhioT1DM dataset

PH	Method	RMSE (mg/dL)	MAE (mg/dL)	gRMSE (mg/dL)
30 minutes	E3NN	$\mathbf{18.92 \pm 2.12}$	$\mathbf{13.46 \pm 1.49}$	$\mathbf{23.40 \pm 2.86}$
	TCN	$20.23 \pm 2.35^{\ddagger}$	$14.59 \pm 1.66^{\ddagger}$	$25.04 \pm 2.90^{\ddagger}$
	CRNN	$21.48 \pm 2.63^{\ddagger}$	$15.80 \pm 2.03^{\ddagger}$	$27.25 \pm 3.28^{\ddagger}$
	LSTM	20.11 ± 2.48	$14.06 \pm 1.69^{\dagger}$	$24.84 \pm 2.88^{*}$
	Bi-LSTM	$20.15 \pm 2.25^{*}$	$14.16 \pm 1.63^{\ddagger}$	$25.01 \pm 2.73^{\ddagger}$
	SVR	$21.37 \pm 2.25^{\ddagger}$	$16.27 \pm 1.68^{\ddagger}$	$26.73 \pm 2.87^{\ddagger}$
	ARIMA	$20.43 \pm 2.19^{\ddagger}$	$14.42 \pm 1.41^{\ddagger}$	$24.51 \pm 2.65^{\ddagger}$
60 minutes	E3NN	$\mathbf{32.54 \pm 3.61}$	$\mathbf{24.05 \pm 2.94}$	$\mathbf{41.52 \pm 4.83}$
	TCN	$34.21 \pm 3.71^{\dagger}$	$25.29 \pm 2.99^{\ddagger}$	$44.26 \pm 4.79^{\ddagger}$
	CRNN	$34.05 \pm 4.26^{\ddagger}$	$25.57 \pm 3.60^{\ddagger}$	$44.22 \pm 5.57^{\ddagger}$
	LSTM	$33.10 \pm 3.84^{*}$	24.50 ± 3.08	$42.65 \pm 5.20^{*}$
	Bi-LSTM	$33.76 \pm 4.06^{\ddagger}$	$25.10 \pm 3.31^{\dagger}$	$43.87 \pm 5.20^{\ddagger}$
	SVR	$33.99 \pm 3.59^{\ddagger}$	$25.69 \pm 2.77^{\ddagger}$	$44.21 \pm 4.94^{\ddagger}$
	ARIMA	$35.51 \pm 3.72^{\ddagger}$	$26.03 \pm 2.69^{\ddagger}$	$43.89 \pm 4.65^{*}$

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

noted that the performance of ARIMA is good for the 30-minute PH but degraded with a longer PH. A possible explanation is that the ARIMA method uses a linear equation, for which it is difficult to capture non-linear long-term temporal dependencies. The trajectories of the CGM measurements and the predictive results of the E3NN, LSTM, TCN, and ARIMA methods are shown in Figure 6.6. The dashed green and cyan lines indicate the thresholds of hypoglycaemia and hyperglycaemia, respectively. When compared with the LSTM and TCN methods, the E3NN method obtained less underestimation in hyperglycaemic regions and less overestimation in hypoglycaemic regions. However, it is observed that deep learning methods lack sensitivity for BG changes at the troughs of the plotted curves, as highlighted by the black ellipses. This may cause missed detection of severe hypoglycaemia and lead to life-threatening events in clinical settings. Therefore, we introduced the corresponding lower bounds to address this challenge.

It is worth noting that the performance of E3NN is comparable to that of models proposed in Chapter 3, which use exogenous input features obtained from meal and bolus

Table 6.2: Performance of the prediction models evaluated on the ABC4D dataset

PH	Method	RMSE (mg/dL)	MAE (mg/dL)	gRMSE (mg/dL)
30 minutes	E3NN	$\mathbf{20.11 \pm 2.54}$	$\mathbf{14.34 \pm 1.78}$	$\mathbf{24.90 \pm 3.39}$
	TCN	21.86 ± 5.52	$15.06 \pm 1.89^{\ddagger}$	27.05 ± 6.33
	CRNN	$22.96 \pm 3.28^{\ddagger}$	$16.61 \pm 2.21^{\ddagger}$	$29.11 \pm 4.35^{\ddagger}$
	LSTM	$20.26 \pm 2.58^{\ddagger}$	$14.53 \pm 1.84^{\ddagger}$	$25.16 \pm 3.37^{\dagger}$
	Bi-LSTM	$20.36 \pm 2.56^{\ddagger}$	$14.64 \pm 1.84^{\ddagger}$	$25.38 \pm 3.33^{\ddagger}$
	SVR	$21.89 \pm 2.52^{\ddagger}$	$16.64 \pm 1.99^{\ddagger}$	$27.74 \pm 3.56^{\ddagger}$
	ARIMA	$22.15 \pm 2.59^{\ddagger}$	$15.61 \pm 1.90^{\ddagger}$	$26.48 \pm 3.56^{\ddagger}$
60 minutes	E3NN	$\mathbf{33.88 \pm 4.81}$	$\mathbf{24.98 \pm 3.56}$	$\mathbf{43.77 \pm 6.44}$
	TCN	40.56 ± 17.10	$26.17 \pm 3.88^{\ddagger}$	51.30 ± 19.48
	CRNN	38.23 ± 13.61	$26.97 \pm 4.19^{\ddagger}$	49.14 ± 15.07
	LSTM	34.31 ± 4.94	$25.36 \pm 3.67^{\ddagger}$	44.32 ± 6.80
	Bi-LSTM	$34.38 \pm 5.15^{*}$	$25.43 \pm 3.79^{\ddagger}$	$44.48 \pm 7.11^{*}$
	SVR	$34.90 \pm 4.76^{\ddagger}$	$26.46 \pm 3.61^{\ddagger}$	$45.43 \pm 6.55^{\ddagger}$
	ARIMA	$38.59 \pm 5.12^{\ddagger}$	$28.02 \pm 3.74^{\ddagger}$	$47.95 \pm 7.14^{\ddagger}$

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

entries and real-time wristband measurements. In this context, there is a trade-off between the improvement of RMSE and the use of additional medical devices. If the applications in T1D management do not require extremely accurate BG prediction, these devices might not be necessary.

Comparison Among Deep Learning Methods

As an edge AI application implemented on a hardware platform with limited computational resources, the memory footprint and operations per inference, as well as prediction accuracy, are important considerations during the selection of deep learning models. The deep learning models were developed by the TensorFlow library. Thus, we converted them into a TensorFlow Lite compressed format that supports on-device inference for many mobile and Internet of things devices, to analyze the hardware requirements. Table 6.4 summarises the number of parameters (Param) and floating-point operations per second (FLOPS), peak SRAM, Flash, and the error score (ES) that sums up the RMSE and MAE

Table 6.3: Performance of the prediction models evaluated on the ARISES dataset

PH	Method	RMSE (mg/dL)	MAE (mg/dL)	gRMSE (mg/dL)
30 minutes	E3NN	$\mathbf{20.45 \pm 3.81}$	$\mathbf{14.78 \pm 2.62}$	$\mathbf{25.31 \pm 5.09}$
	TCN	$22.01 \pm 4.19^{\ddagger}$	$15.98 \pm 2.91^{\ddagger}$	$28.03 \pm 5.80^{\ddagger}$
	CRNN	$24.43 \pm 5.19^{\ddagger}$	$17.93 \pm 3.76^{\ddagger}$	$31.67 \pm 7.28^{\ddagger}$
	LSTM	20.74 ± 3.66	15.03 ± 2.55	$26.09 \pm 4.89^{*}$
	Bi-LSTM	$20.86 \pm 3.78^{*}$	$15.19 \pm 2.73^{\ddagger}$	$26.30 \pm 5.12^{\dagger}$
	SVR	$22.87 \pm 3.99^{\ddagger}$	$17.25 \pm 2.99^{\ddagger}$	$29.10 \pm 5.49^{\ddagger}$
	ARIMA	$21.76 \pm 4.73^{\ddagger}$	$15.59 \pm 2.71^{\ddagger}$	$26.20 \pm 5.20^{\ddagger}$
60 minutes	E3NN	$\mathbf{35.55 \pm 7.24}$	$\mathbf{26.22 \pm 5.28}$	$\mathbf{46.37 \pm 10.11}$
	TCN	$37.01 \pm 7.72^{\ddagger}$	$27.64 \pm 5.66^{\ddagger}$	$48.65 \pm 10.76^{\ddagger}$
	CRNN	$38.07 \pm 8.20^{\ddagger}$	$28.49 \pm 6.15^{\ddagger}$	$50.17 \pm 11.37^{\ddagger}$
	LSTM	$36.68 \pm 6.97^{\ddagger}$	$27.02 \pm 5.12^{*}$	$48.80 \pm 9.83^{\ddagger}$
	Bi-LSTM	$37.14 \pm 7.38^{\ddagger}$	$27.59 \pm 5.45^{\ddagger}$	$49.00 \pm 10.55^{\ddagger}$
	SVR	$37.08 \pm 7.48^{\ddagger}$	$27.75 \pm 5.44^{\ddagger}$	$48.79 \pm 10.42^{\ddagger}$
	ARIMA	$39.51 \pm 8.16^{\ddagger}$	$28.75 \pm 5.65^{\ddagger}$	$49.73 \pm 11.00^{\ddagger}$

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

Table 6.4: Comparison between the proposed E3NN and considered deep learning baseline methods.

Method	Param	FLOPs	SRAM	Flash	ES (mg/dL)
TCN	124K	248K	13.7KB	499KB	94.44
CRNN	52K	136K	8.1KB	227KB	96.78
LSTM	53k	1577K	7.3KB	2096KB	91.87
Bi-LSTM	141K	412K	13.2KB	624KB	93.17
E3NN	**32K**	**93K**	13.8KB	**171KB**	**88.97**

for the 30-minute and 60-minute PHs. It is noteworthy that the E3NN model obtained the best prediction performance (the lowest ES) with the smallest numbers of parameters, FLOPs, and Flash. Although the E3NN consumes relatively high peak SRAM, this amount is much smaller than the available capacity of most commercial microcontroller units, as well as the target SoC in this work (64KB). We observe that the LSTM model achieved the second-best ES at the cost of a large number of parameters and Flash requirement that is likely to exceed the memory constraint.

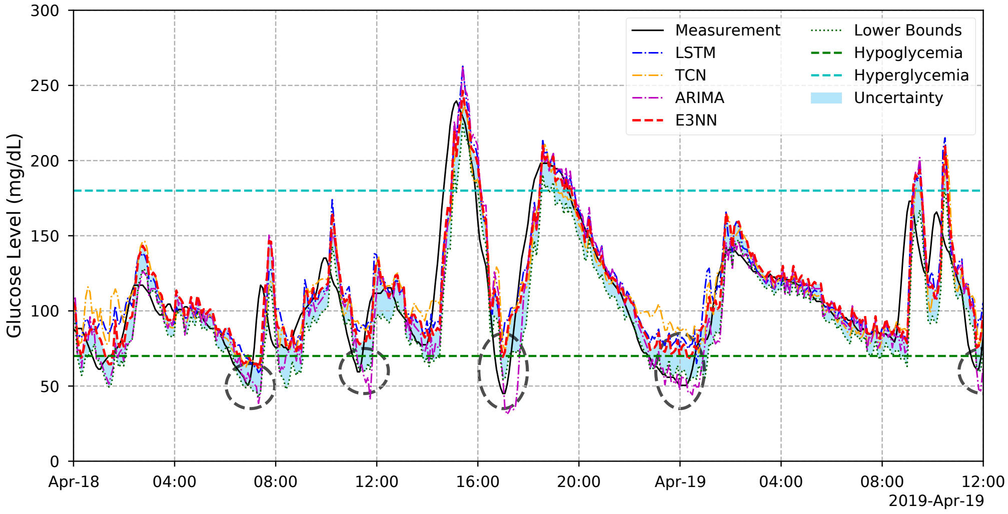


Figure 6.6: 1.5-day prediction performance of the considered methods on the ARISES dataset with a 30-minute PH. The solid black line indicates the actual CGM measurements, while the dash-dotted blue, orange, and magenta lines are the results of LSTM, TCN, ARIMA methods. The dashed red line indicates the results of the E3NN method, where the lower bounds and uncertainty are represented by the dotted green line and shaded blue area, respectively. The black ellipses highlight the hypoglycaemic events detected by the lower bounds.

Hypoglycaemia Detection

A widespread application of BG prediction in T1D management systems is to prevent hypoglycaemic episodes that would lead to fatal complications. We detected impending hypoglycaemia using the lower bounds of E3NN predictions and the prediction values of the considered baseline methods at the same PHs. Table 6.5 presents the MCC scores evaluated on the three clinical datasets. Although the TCN-based model obtained higher RMSE, MAE, and gRMSE results than the LSTM and Bi-LSTM in Table 6.1, 6.2, and 6.3, it is worth noting that the TCN achieved the best performance of hypoglycaemia detection among all the considered deep learning baseline methods. A possible explanation is that the TCN-based models have longer effective memory than canonical RNNs with the same capacity, as suggested in [289]. Therefore, the TCN model could better understand the patterns of hypoglycaemia caused by external events that occurred hours ago, such as postprandial hypoglycaemia. In our previous work [101], we also noticed that the TCN model exhibited a short prediction time lag, indicating good sensitivity to the changes

Table 6.5: MCC scores of the hypoglycaemia prediction evaluated on the three datasets

PH	Method	OhioT1DM	ABC4D	ARISES
30 min	E3NN	$\mathbf{0.70 \pm 0.09}$	$\mathbf{0.68 \pm 0.09}$	$\mathbf{0.70 \pm 0.12}$
	TCN	$0.55 \pm 0.10^{\ddagger}$	$0.49 \pm 0.20^{\ddagger}$	$0.40 \pm 0.12^{\ddagger}$
	ARIMA	$0.65 \pm 0.09^{*}$	0.59 ± 0.07	0.61 ± 0.10
60 min	E3NN	$\mathbf{0.57 \pm 0.09}$	$\mathbf{0.54 \pm 0.11}$	$\mathbf{0.49 \pm 0.14}$
	TCN	$0.38 \pm 0.11^{\ddagger}$	$0.37 \pm 0.18^{\ddagger}$	$0.30 \pm 0.15^{\ddagger}$
	ARIMA	0.49 ± 0.06	$0.48 \pm 0.06^{\ddagger}$	0.45 ± 0.12

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

in the troughs of glucose trajectories, i.e., hypoglycaemia regions. Meanwhile, it is noted that the ARIMA outperformed the SVR model with a higher MCC score. Therefore, we compared the E3NN with the TCN model and ARIMA in Table 6.5.

Notably, the E3NN model achieved the highest MCC scores for each dataset in both 30-minute and 60-minute PHs. It is interesting to note that the MCC scores of the ARIMA method are significantly higher than the TCN and the other deep learning methods, while the improvement of the E3NN on the ABC4D and ARISES datasets is not significant when compared with the ARIMA. In Fig 6.6, we see that the ARIMA predictions can identify more hypoglycaemic events than the TCN with a time-shifted delay on the curve, which, however, degrades the RMSE performance (Table 6.1). It is reasonable since the weights of the DNN models were optimised by the regression loss that aims to enhance RMSE performance, instead of the accuracy of hypoglycaemia detection. It is observed that the lower bounds of the E3NN curve successfully detected five hypoglycaemic events circled by the black ellipse, which are likely to be missed if we use the prediction values only. In particular, the use of lower bounds increased average MCC scores for the three datasets by 0.13 ($p < 0.005$) and 0.17 ($p < 0.005$) for the 30-minute and 60-minute PHs, respectively. Hence, these results suggest that evidential regression is an important improvement in BG prediction methods based on deep learning.

6.7.2 Edge Implementation

The proposed RNN predictor was implemented on the BLE SoC for edge computing. By means of utilising the optimised CMSIS-DSP library that is pre-compiled and included in the latest NRF52 SDK, the SoC is able to accept high-throughput data while performing rapid computation of matrix operations that typically involve single-cycle multiplication and accumulation. This enables efficient data processing with minimal overhead and the real-time execution of computation-intensive algorithms.

Table 6.6 presents the detailed utilization of Flash and SRAM memory in Byte (B) for the implementation of E3NN. As the SRAM memory was allocated dynamically during the run time, the implementation of this RNN model only led to an increase of 2.48% on the SRAM utilization compared with that without the edge computation. Whereas the capacity of the Flash memory is the main bottleneck that limits the size of the RNN predictor, occupying 66.13% of the total flash utilization. In addition, the computation time was empirically estimated by executing each layer for 100 rounds and averaging the run time through the UART timestamp. The result shows an average computation time of approximately 500 ms. Moreover, compared with the implementation by TensorFlow Lite Micro in Table 6.4, our implementation significantly reduced Flash from 171 KB to 125 KB and peak SRAM from 13.8 KB to 3.5 KB, mainly because it computed outcomes using low-level CMSIS-DSP APIs without interpreting the network graph. For each considered T1D subject, the RMSE between the testing results of Python model and those of the edge model is less than 10^{-5} mg/dL.

The final firmware for the BLE SoC utilises 189.03 KB Flash and 16.12 KB SRAM memory, which provides the following six functionalities: 1) CGM sensor connectivity and readout, 2) input data pre-processing, 3) edge computing of the RNN predictor, 4) external flash memory management, 5) basic user interactions, and 6) developer mode for data readout and parameter update.

Table 6.6: Details of Flash and SRAM Memory Footprint

Layer	Input Shape	Flash (B)	SRAM (B)	Time
Input	(2, 12)	0	96	0
GRU 1 **	(1, 2)	52,224	1,536	22.58 ms
GRU 2 **	(1, 64)	37,632	768	16.16 ms
Attention	(12, 32)	20,480	2,096	28.92 ms
Dense	(1, 64)	16,640	256	6.22 ms
Evidential	(1, 64)	1,040	16	0.32 ms
Output	(1, 4)	0	16	0

** This layer is repeatedly executed for 12 times.

6.7.3 Power Analysis

Power estimation was conducted by using a source meter Keithley 2606A, which supplied 3 volts and monitored the power in real time. Figure 6.7 presents the power monitoring of a typical cycle that lasts for 13 seconds, which consumes an average run-time power of 3.78 mW. The power spikes at the initial stage indicate the Bluetooth scanning process, which involves a tunable window and interval. The highest power occurring in the middle indicates the edge computing for the embedded predictor. During the pulse at the end, the system polls the power-gating circuit to enter shutdown mode.

At the very beginning, the device keeps scanning the target sensor and involves an on-off current switching with a peak value around 6.5 mA. To reduce the power consumption of this process, the BLE scanning window is shortened into a duty cycle of 10%, resulting in an average power of 5 mW. Once the target sensor is connected, the BLE SoC will start the authentication and bonding process, which typically lasts for 2 seconds. After the success of bonding, the device is able to request the glucose data and start prediction. Notably, the running of the RNN predictor is the most energy-hungry process as it utilises the on-chip digital signal processing for the computation of floating-point arithmetic operations, but only takes a short operating time of around 500 ms. Depending on the predicted blood glucose level, the notifications of the low excursion with intermittent alarms are generated through the LED and buzzer. During this period, the system waits for user response but maintains a low power that is less than 1 mW. If the button is pressed, i.e., the warning of a hypoglycaemic event is confirmed, the system will enter

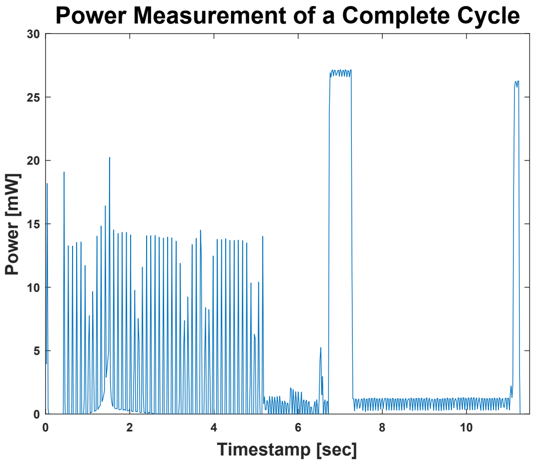


Figure 6.7: Power measurement of a complete run cycle that involves BLE scanning, authentication and bonding, edge AI computing, and hypoglycaemia notification. Among these processes, the two main contributors, including BLE scanning and edge AI computation, lasted for 5 and 0.52 seconds, which consumed an average power of 5 mW and 27 mW, respectively.

the shutdown mode by triggering the on-board timer for power gating.

In the real application, the device can be powered by a single coin battery, e.g., CR2032, which typically has a capacity of 240 mAh. This capacity enables the device to operate for six months, assuming that each CGM readout is processed every five minutes. The commercial CGM sensors and transmitters in the market typically require a replacement every 10 days and three months, respectively. Thus, the achieved battery life of our wearable device is long enough to cover these periods.

6.7.4 In Silico Trial

To evaluate the performance of the whole system with the wearable device, we performed a 3-month hardware-in-the-loop *in silico* trial using the UVA/Padova T1D simulator, which is a common experimental setup of pre-clinical trials in T1D management systems. In particular, we employed 10 virtual adult subjects with additional intra- and inter-subject variability [264] and used the carbohydrate of meal protocol as follows: 70 g (breakfast, 7 am), 110 g (lunch, 2 pm), and 90 g (dinner, 9 pm), with the variability of mealtime (STD = 30%) and meal size (CV = 10%). The simulator sent CGM values to the wearable device

Table 6.7: Glycaemic outcomes of the in silico trial

Method	TIR (%)	TBR (%)	TSH (%)	LBGI
Control	74.26 ± 7.62	$5.44 \pm 3.38^{\ddagger}$	$2.00 \pm 1.45^{\ddagger}$	$1.50 \pm 0.81^{\ddagger}$
PLGM	74.83 ± 9.02	2.02 ± 1.14	0.47 ± 0.38	0.65 ± 0.28

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

and received 30-minute predictions through a debug mode with the UART and USB ports. We performed the PLGM algorithm with the settings in [189], where the pump suspended basal insulin when the predictions were at or below the threshold of hypoglycaemia, i.e., 70 mg/dL.

Table 6.7 presents the outcomes of PLGM and a control group (i.e., no suspension) as a baseline, evaluated by TIR of [70, 180] mg/dL, TBR (BG< 70 mg/dL), time of severe hypoglycaemia (TSH) (BG< 54 mg/dL), LBGI. It is noted that integrating the wearable device with PLGM significantly reduced the LBGI and percent time of hypoglycaemia and severe hypoglycaemia without a decrease of TIR. Figure 6.8 depicts the outcomes of CVGA for a virtual adult subject. We observe that, compared with the control group, more of the PLGM dots are located in the left bottom zones. Specifically, the PLGM improved the percentage of the A+B zone from 67% to 77% and reduced 10% of the dots in the D+E zone, indicating good BG control. Besides the PLGM, other interventions, such as glucagon delivery and rescue carbohydrate recommendations, could also be performed in clinical settings to further reduce the incidence of hypoglycaemia, based on the real-time BG predictions of the wearable device.

6.8 Conclusion

In this chapter, we developed E3NN based on the FCNN framework for BG prediction and a novel IoMT-enabled wearable device to implement the deep learning algorithm for real-time BG prediction and hypoglycaemia warning with edge computing on the SoC. When evaluated on the three clinical datasets, the proposed model obtained the best prediction accuracy for both future BG level and impending hypoglycaemic events with

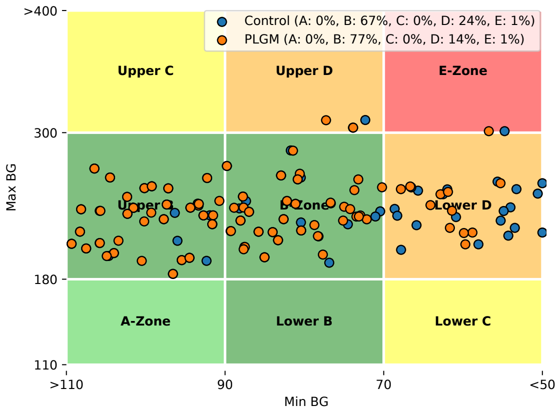


Figure 6.8: CVGA plot comparing PLGM (orange dots) against the control group (blue dots) for a virtual adult subject in the trial. Each dot stands for the extreme values of BG trajectories over 24 hours.

the smallest number of model parameters and FLOPs, compared with the considered deep learning baseline methods. Moreover, the optimised hardware design of the wearable device enables extremely low energy consumption for edge inference and BLE connectivity, which can run 24/7 operations over six months. The results of hardware-in-the-loop *in silico* trials demonstrated that integrating the wearable device into the T1D management system can improve glycaemic outcomes of BG control.

Chapter 7

Conclusion

This thesis investigated deep learning applications in diabetes management and demonstrated that the proposed deep learning-based methods have achieved state-of-the-art performance for critical decision support in multiple clinical scenarios. Taking advantage of the substantial physiological data generated by wearable devices and daily entries recorded by smartphone apps, we have proposed novel deep learning frameworks in glucose prediction for proactive interventions, and in glycaemic control to maintain BG levels in a therapeutically appropriate range. To take a further step towards the clinical use of the proposed algorithms, we have proposed GluGAN to generate synthetic BG data for data augmentation in developing data-driven algorithms and ambulatory glucose profiles for T1D subjects on SMBG therapy, and an IoMT framework to facilitate people with diabetes and clinicians to visualise historical glucose profiles, store clinical data, develop and implement deep learning models with little engineering work.

7.1 Contributions

Chapter 2 introduced the background of diabetes management and provided an overview of deep learning technology. Following the preferred reporting items for systematic review and meta-analyses approach, we performed a systematic literature review by searching the journal articles that were published between January 1, 2016 and September 1, 2022 in three reputable online databases. A comprehensive literature summary with extracted key information, including application cases, data sources, development process, main outcomes, baselines, and limitations, was presented. The challenges in medical data and deep learning models were identified.

Chapter 3 provided a novel deep learning framework, FCNN, for BG prediction and addressed the challenges of model confidence and data availability. We developed a bidirectional GRU model with a modified many-to-one attention mechanism to improve the prediction accuracy and evidential learning that provides theoretically supported CIs. MAML with the first-order approximation was employed to enable fast adaptation with limited data and solved the cold-start issue. The proposed model was evaluated on the OhioT1DM dataset, ARISES dataset, and ABC4D dataset and compared with a group of machine learning and deep learning baseline methods in terms of BG level prediction, hypoglycaemia prediction, and fast adaptation performance. In the second part of Chapter 3, we investigated the efficacy of physiological data measured by the wearable wristband sensor (Empatica E4) in diabetes management, which were collected in our previous trial with 12 T1D participants and provided in the ARISES dataset. We first analysed the association between the sensor data and adverse glycaemic events through mixed effects logistic regression. Then, the sensor data were used in the FCNN framework to build a deep learning model to predict BG levels, hyperglycaemia, and hypoglycaemia over PHs from 15 to 60 minutes.

Chapter 4 presented three novel DRL applications in glycaemic control. First, a DRL model based on double Q-learning and DRNNs was proposed to optimise single-hormone (basal insulin) and dual-hormone (basal insulin and glucagon) delivery in AP systems. The personalised control algorithms were developed by means of a novel two-step learning framework that accelerated model training and can improve safety in clinical settings. The model was evaluated in the UVA/Padova T1D simulator with 10 virtual adults and 10 virtual and compared with the standard LGS. Secondly, we proposed a DRL-based meal insulin bolus advisor for both MDI and insulin pump therapy using the actor-critic DDPG architecture. The model was trained by the two-step learning framework and prioritised memory replay, and evaluated in the UVA/Padova T1D simulator as well. The model performance was compared with that of SBC. Finally, we developed an offline DRL and OPE framework for basal insulin delivery and removed the need for long-term error and trial exploration in simulators or actual clinical trials, enabling the DRL models to be trained, validated, and tested in a safe and offline setting. The proposed DRL model was based on TD3 and behaviour cloning regularization, while FQE was used for OPE. The proposed framework was validated on both the offline *in silico* dataset generated by the UVA/Padova T1D simulator and the clinical OhioT1DM datasets with 12 T1D adults.

Chapter 5 presented a novel deep learning framework, GluGAN, to generate personalised BG time series data. GluGAN incorporated three additional DNN modules, including an embedding network, a recovery network, and a supervisor network, into a standard GAN model. Leveraging unsupervised learning (i.e., adversarial learning) and supervised learning, the model was shown to learn similar distribution densities and autoregressive relationships to those of real CGM data. We evaluated the performance of GluGAN on the OhioT1DM dataset, ARISES dataset, and ABC4D dataset and compared it with four GAN-based baseline methods for time series generation. GluGAN was further applied to data augmentation in BG prediction with TRTR and TRTR routines. The experiments were performed with three standard data-driven predictors.

Chapter 6 provided an IoMT-enabled wearable device with a low-cost and power-efficient SoC that communicated with CGM and other devices of T1D management through BLE and performed edge computing for the model inference of the embedded deep learning algorithm. A cloud platform and a desktop platform were developed for model training and data backup. An iOS smartphone app was designed to record daily events, visualise BG trajectories, and provide decision support, according to the feedback of T1D users in the ARISES clinical trial. Utilising the FCNN framework proposed in Chapter 3, an embedded model, E3NN, was developed for real-time BG prediction and hypoglycaemia detection with CGM measurements on edge devices. The embedded model was evaluated by the OhioT1DM dataset, ARISES dataset, and ABC4D dataset and compared against a variety of machine learning and deep learning baseline methods.

We analyzed the power and memory footprint of the wearable device. A hardware-in-the-loop *in silico* trial using the UVA/Padova T1D simulator was performed to validate the therapeutic efficacy of the PLGM system integrated with the proposed wearable device.

7.2 Recommendations for Future Work

This thesis has presented pioneering research on deep learning applications to diabetes management and provided new paradigms for BG prediction, glycaemic control, synthetic BG data generation, and digital health systems. As identified in Chapter 2, the number of research in these areas is increasing exponentially, and deep learning is evolving more rapidly than ever before. We expect that deep learning technology will be widespread in clinical settings in the future, and the potential research directions are presented as follows.

7.2.1 Non-Invasive BG Monitoring and Prediction With Wearable Wristband Sensors

In Chapter 3, we have demonstrated that several physiological features measured by the wristband sensor (Empatica E4), especially heart rate variability, were significantly associated with hypoglycaemia and hyperglycaemia and improved BG prediction. A very recent study has detected current hypo- and hyperglycaemic events using a machine learning algorithm (XGBoost) and wristband sensor data [172]. Therefore, there is also a possibility to monitor current BG levels and predict upcoming hypoglycaemia and hyperglycaemia purely based on non-invasive physiological data. The success of this application will enable people with severe T2D or T1D to discontinue CGM therapy, but SMBG might be required to calibrate the algorithm. To achieve this target, future work will include developing powerful time series models with the recent advances in deep learning. In particular, self-attention models (i.e., transformers) will be investigated. We used the encoder of a standard transformer as a baseline method in Chapter 3, which achieved comparable performance. To better learn the temporal dynamics of multivariate time series and reduce the complexity of self-attention mechanisms, many recent variants of the transformer, such as temporal fusion transformers [290], will be explored.

Wristband sensors, such as Empatica E4, are quite sensitive to motion artefacts, so it is difficult to obtain accurate measurements with too many hand movements [291]. Thus, an algorithm to detect exercise and reduce measurement error for the wristband will be developed. In fact, it would also be possible to design a customised wristband to measure certain non-invasive physiological features and provide on-device BG monitoring, leveraging the edge computing for deep learning proposed in Chapter 6. In future work, it would be more appropriate for a deep learning model to disentangle the effect of different features, especially meal intake and insulin delivery, before it can be used in clinical settings. There are two potential solutions for future work. One is to introduce monotonic constraints in the DNNs to specify the insulin's negative effect on BG levels, such as restricting the layer weights of shallow networks [292] and training with heuristic regularizations [293]. The other is to incorporate physiological models to process these events, such as composite minimal models of glucose regulation [188].

7.2.2 Glycaemic Control With DRL and Data of Physical Activities

Despite being able to reliably model glucose-insulin dynamics in T1D, the current FDA-accepted version of the UVA/Padova simulator [81] lacks the effect of physical activity and health conditions (e.g., recurrent illness) that are known to significantly influence insulin sensitivity in people with T1D. In particular, the effect of physical activities has been proven to be very complicated to model. Thus, the modelling of the insulin-mediated and non-insulin-mediated effect on muscle glucose need to be further assessed and developed through more research [294].

In future work, we will first incorporate physical activity with the double DQN and DDPG algorithms, which have been demonstrated to significantly glycaemic control in Chapter 4. If physical activity, basal insulin, and the data of other daily events become available in future public CGM datasets, the offline DRL and OPE framework will be further investigated. There is rapid development in DRL, and we plan to explore the latest advances in this area, such as combining DRL and meta-learning [295]. For all the proposed methods, *in silico* experiments with more realistic variability need to be conducted before the clinical use of the offline DRL algorithm, for which a new version of the UVA/Padova T1D simulator (S2017) [265] will be considered. An alternative solution is to develop control algorithms coped with unannounced physical activities [232].

7.2.3 GAN-Based Personalised Diabetes Simulator and Digital Twins

By using conditional inputs of SMBG measurements, carbohydrates from meal intake, and insulin delivery to generation BG time series, we presented a pilot study on the personalised data-driven T1D simulator in Chapter 5. However, to build a fully functional personalised diabetes simulator, the interpretability and causality of deep learning models have to be improved. For instance, we need to make sure that the synthetic postprandial BG levels will rise if the subject increases carbohydrate intake without adjusting insulin therapy. Therefore, future work will include casual graphs [296] and exponential objective functions [297]. Furthermore, we consider combining physiological models and CGM models (e.g, the UVA/Padova models [81] or the Hovorka models [215]) with GluGAN to obtain robust generative performance and finally develop personalised T1D digital twins.

The digital twins will initiate revolutionary changes in both pre-clinical and clinical T1D studies, which allows clinicians to freely test and adjust treatment and daily management policy in a virtual environment. For the offline DRL algorithms in Chapter 4, although policy values are highly correlated with the occurrence of adverse glycaemic events, there is a lack of an explicit relationship between OPE metrics and specific BG trajectories. In this context, the data-driven personalised simulators can be integrated with OPE methods to estimate the efficacy of the proposed AP controllers.

7.2.4 On-Device Decision Support With CGM Transmitters and Insulin Pumps

In Chapter 6, the edge computing for the E3NN is based on a tiny microcontroller unit (nRF52832) with a dimension of 3.0 × 3.2 mm in wafer-level chip scale packaging. It is worth noting that nRF52832 is a popular microcontroller unit that supports advanced

BLE features, which has been used in recent commercialised CGM and insulin pumps. Therefore, in future work, the deep learning decision support models will be deployed in other T1D IoMT wearable devices to provide on-device decision support. With the collaboration of manufacturers, it will be possible to implement the algorithms on CGM transmitters and insulin pumps. The size of CGM transmitters is around 5.0×3.0 cm (Dexcom G6), while insulin pumps measure approximately 10.0×5.0 cm (Medtronic MiniMed 640G). There should be enough space to integrate external Flash memory or an additional microcontroller unit into current SoCs for the embedded deep learning algorithms. Such implementation will reduce the need for these wearable devices to constantly stay connected with external controllers, and thus largely relieve daily burdens for people with diabetes.

For the proposed IoMT system and devices, further clinical trials will be performed to investigate the performance of software and hardware in real-world settings and modify the functions and GUIs according to user feedback.

Bibliography

[1] P. B. Jensen, L. J. Jensen, and S. Brunak, "Mining electronic health records: towards better research applications and clinical care," *Nature Reviews Genetics*, vol. 13, no. 6, p. 395, 2012.

[2] C. M. Bishop, *Pattern recognition and machine learning.* springer, 2006.

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

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

[5] World Health Organization, *Global report on diabetes.* World Health Organization, 2016.

[6] J. C. Pickup, "Management of diabetes mellitus: is the pump mightier than the pen?" *Nature Reviews Endocrinology*, vol. 8, no. 7, p. 425, 2012.

[7] A. D. Deshpande, M. Harris-Hayes, and M. Schootman, "Epidemiology of diabetes and diabetes-related complications," *Physical Therapy*, vol. 88, no. 11, pp. 1254–1264, 2008.

[8] R. D. Coffen and L. M. Dahlquist, “Magnitude of type 1 diabetes self-management in youth health care needs diabetes educators,” *The Diabetes Educator*, vol. 35, no. 2, pp. 302–308, 2009.

[9] J. Vora and T. Heise, “Variability of glucose-lowering effect as a limiting factor in optimizing basal insulin therapy: a review,” *Diabetes, Obesity and Metabolism*, vol. 15, no. 8, pp. 701–712, 2013.

[10] L. Heinemann, G. Freckmann, D. Ehrmann, G. Faber-Heinemann, S. Guerra, D. Waldenmaier, and N. Hermanns, “Real-time continuous glucose monitoring in adults with type 1 diabetes and impaired hypoglycaemia awareness or severe hypoglycaemia treated with multiple daily insulin injections (HypoDE): a multicentre, randomised controlled trial,” *The Lancet*, vol. 391, no. 10128, pp. 1367–1377, 2018.

[11] D. S. Feig, L. E. Donovan, R. Corcoy, K. E. Murphy, S. A. Amiel, K. F. Hunt, E. Asztalos, J. F. Barrett, J. J. Sanchez, A. De Leiva *et al.*, “Continuous glucose monitoring in pregnant women with type 1 diabetes (CONCEPTT): a multicentre international randomised controlled trial,” *The Lancet*, vol. 390, no. 10110, pp. 2347–2359, 2017.

[12] P. Herrero, P. Georgiou, N. Oliver, D. G. Johnston, and C. Toumazou, “A bio-inspired glucose controller based on pancreatic β-cell physiology,” *Journal of Diabetes Science and Technology*, vol. 6, no. 3, pp. 606–616, 2012.

[13] N. Oliver, M. Reddy, C. Marriott, T. Walker, and L. Heinemann, “Open source automated insulin delivery: addressing the challenge,” *npj Digital Medicine*, vol. 2, no. 1, pp. 1–5, 2019.

[14] A. Slomski, “Artificial pancreas improves glycemic control,” *JAMA*, vol. 320, no. 20, pp. 2068–2068, 2018.

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

[16] E. A. Ryan, J. Holland, E. Stroulia, B. Bazelli, S. A. Babwik, H. Li, P. Senior, and R. Greiner, "Improved A1C levels in type 1 diabetes with smartphone app use," *Canadian Journal of Diabetes*, vol. 41, no. 1, pp. 33–40, 2017.

[17] F. L. Schwartz, C. R. Marling, and R. C. Bunescu, "The promise and perils of wearable physiological sensors for diabetes management," *Journal of Diabetes Science and Technology*, vol. 12, no. 3, pp. 587–591, 2018.

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

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

[20] A. L. Carlson, D. M. Mullen, and R. M. Bergenstal, "Clinical use of continuous glucose monitoring in adults with type 2 diabetes," *Diabetes Technology & Therapeutics*, vol. 19, no. S2, pp. S–4, 2017.

[21] American Diabetes Association and others, "2. classification and diagnosis of diabetes," *Diabetes Care*, vol. 40, no. Supplement 1, pp. S11–S24, 2017.

[22] T. Tuomi, N. Santoro, S. Caprio, M. Cai, J. Weng, and L. Groop, "The many faces of diabetes: a disease with increasing heterogeneity," *The Lancet*, vol. 383, no. 9922, pp. 1084–1094, 2014.

[23] A. Katsarou, S. Gudbjörnsdottir, A. Rawshani, D. Dabelea, E. Bonifacio, B. J. Anderson, L. M. Jacobsen, D. A. Schatz, and Å. Lernmark, "Type 1 diabetes mellitus," *Nature Reviews Disease Primers*, vol. 3, no. 1, pp. 1–17, 2017.

[24] Centers for Disease Control and Prevention and others, "National diabetes fact sheet: national estimates and general information on diabetes and prediabetes in the united states, 2011," *Atlanta: US Department of Health and Human Services*, vol. 201, no. 1, pp. 2568–2569, 2011.

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

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

[27] M. Sussman, J. Benner, M. J. Haller, M. Rewers, and R. Griffiths, "Estimated lifetime economic burden of type 1 diabetes," *Diabetes Technology & Therapeutics*, vol. 22, no. 2, pp. 121–130, 2020.

[28] W. H. Polonsky, D. Hessler, K. J. Ruedy, and R. W. Beck, "The impact of continuous glucose monitoring on markers of quality of life in adults with type 1 diabetes: further findings from the DIAMOND randomized clinical trial," *Diabetes Care*, vol. 40, no. 6, pp. 736–741, 2017.

[29] S. Clarke and J. Foster, "A history of blood glucose meters and their role in self-monitoring of diabetes mellitus," *British Journal of Biomedical Science*, vol. 69, no. 2, pp. 83–93, 2012.

[30] S. R. Patton and M. A. Clements, "Continuous glucose monitoring versus self-monitoring of blood glucose in children with type 1 diabetes-are there pros and cons for both?" *US Endocrinology*, vol. 8, no. 1, p. 27, 2012.

[31] B. Floyd, P. Chandra, S. Hall, C. Phillips, E. Alema-Mensah, G. Strayhorn, E. O. Ofili, and G. E. Umpierrez, "Comparative analysis of the efficacy of continuous glucose monitoring and self-monitoring of blood glucose in type 1 diabetes mellitus," *Journal of Diabetes Science and Technology*, vol. 6, no. 5, pp. 1094–1102, 2012.

[32] D. C. Klonoff, "Continuous glucose monitoring: roadmap for 21st century diabetes therapy," *Diabetes Care*, vol. 28, no. 5, pp. 1231–1239, 2005.

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

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

[35] P. Rossetti, J. Bondia, J. Vehí, and C. G. Fanelli, "Estimating plasma glucose from interstitial glucose: the issue of calibration algorithms in commercial continuous glucose monitoring devices," *Sensors*, vol. 10, no. 12, pp. 10 936–10 952, 2010.

[36] P. Adolfsson, C. G. Parkin, A. Thomas, and L. G. Krinelke, "Selecting the appropriate continuous glucose monitoring system–a practical approach," *European Endocrinology*, vol. 14, no. 1, p. 24, 2018.

[37] M. Lind, W. Polonsky, I. B. Hirsch, T. Heise, J. Bolinder, S. Dahlqvist, E. Schwarz, A. F. Ólafsdóttir, A. Frid, H. Wedel *et al.*, "Continuous glucose monitoring vs conventional therapy for glycemic control in adults with type 1 diabetes treated with multiple daily insulin injections: the GOLD randomized clinical trial," *JAMA*, vol. 317, no. 4, pp. 379–387, 2017.

[38] R. W. Beck, T. Riddlesworth, K. Ruedy, A. Ahmann, R. Bergenstal, S. Haller, C. Kollman, D. Kruger, J. B. McGill, W. Polonsky *et al.*, "Effect of continuous glucose monitoring on glycemic control in adults with type 1 diabetes using insulin injections: the DIAMOND randomized clinical trial," *JAMA*, vol. 317, no. 4, pp. 371–378, 2017.

[39] C. A. van Beers, J. H. DeVries, S. J. Kleijer, M. M. Smits, P. H. Geelhoed-Duijvestijn, M. H. Kramer, M. Diamant, F. J. Snoek, and E. H. Serné, "Continuous glucose monitoring for patients with type 1 diabetes and impaired awareness of hypoglycaemia (IN CONTROL): a randomised, open-label, crossover trial," *The Lancet Diabetes & Endocrinology*, vol. 4, no. 11, pp. 893–902, 2016.

[40] J. Bondia, E. Dassau, H. Zisser, R. Calm, J. Vehí, L. Jovanovič, and F. J. Doyle III, "Coordinated basal—bolus infusion for tighter postprandial glucose control in insulin pump therapy," *Journal of Diabetes Science and Technology*, vol. 3, no. 1, pp. 89–97, 2009.

[41] J. Pickup and A. Sutton, "Severe hypoglycaemia and glycaemic control in type 1 diabetes: meta-analysis of multiple daily insulin injections compared with continuous subcutaneous insulin infusion," *Diabetic Medicine*, vol. 25, no. 7, pp. 765–774, 2008.

[42] S. S. Hussain and N. Oliver, *Insulin Pumps and Continuous Glucose Monitoring Made Easy.* Edinburgh, Scotland: Elsevier Health Sciences, 2015.

[43] I. Steineck, J. Cederholm, B. Eliasson, A. Rawshani, K. Eeg-Olofsson, A.-M. Svensson, B. Zethelius, T. Avdic, M. Landin-Olsson, J. Jendle *et al.*, "Insulin pump therapy, multiple daily injections, and cardiovascular mortality in 18 168 people with type 1 diabetes: observational study," *BMJ*, vol. 350, 2015.

[44] H. K. Akturk, D. Giordano, A. Champakanath, S. Brackett, S. Garg, and J. Snell-Bergeon, "Long-term real-life glycaemic outcomes with a hybrid closed-loop system compared with sensor-augmented pump therapy in patients with type 1 diabetes," *Diabetes, Obesity and Metabolism*, vol. 22, no. 4, pp. 583–589, 2020.

[45] D. Kariyawasam, C. Morin, K. Casteels, C. Le Tallec, A. Sfez, C. Godot, E. Huneker, N. Garrec, P.-Y. Benhamou, M. Polak *et al.*, "Hybrid closed-loop insulin delivery versus sensor-augmented pump therapy in children aged 6–12 years: a randomised, controlled, cross-over, non-inferiority trial," *The Lancet Digital Health*, vol. 4, no. 3, pp. e158–e168, 2022.

[46] S. A. McAuley, S. Trawley, S. Vogrin, G. M. Ward, S. Fourlanos, C. A. Grills, M. H. Lee, A. M. Alipoor, D. N. O'Neal, N. A. O'Regan *et al.*, "Closed-loop insulin delivery versus sensor-augmented pump therapy in older adults with type 1 diabetes (ORACL): a randomized, crossover trial," *Diabetes Care*, vol. 45, no. 2, pp. 381–390, 2022.

[47] S. J. Russell, F. H. El-Khatib, M. Sinha, K. L. Magyar, K. McKeon, L. G. Goergen, C. Balliro, M. A. Hillard, D. M. Nathan, and E. R. Damiano, "Outpatient glycemic control with a bionic pancreas in type 1 diabetes," *New England Journal of Medicine*, vol. 371, no. 4, pp. 313–325, 2014.

[48] F. H. El-Khatib, C. Balliro, M. A. Hillard, K. L. Magyar, L. Ekhlaspour, M. Sinha, D. Mondesir, A. Esmaeili, C. Hartigan, M. J. Thompson *et al.*, "Home use of a bihormonal bionic pancreas versus insulin pump therapy in adults with type 1 diabetes: a multicentre randomised crossover trial," *The Lancet*, vol. 389, no. 10067, pp. 369–380, 2017.

[49] M. Grady, L. B. Katz, H. Cameron, and B. L. Levy, "Diabetes app-related text messages from health care professionals in conjunction with a new wireless glucose meter with a color range indicator improves glycemic control in patients with type 1 and type 2 diabetes: Randomized controlled trial," *JMIR Diabetes*, vol. 2, no. 2, p. e19, 2017.

[50] J. Doupis, G. Festas, C. Tsilivigos, V. Efthymiou, and A. Kokkinos, "Smartphone-based technology in diabetes management," *Diabetes Therapy*, vol. 11, no. 3, pp. 607–619, 2020.

[51] D. Rhyner, H. Loher, J. Dehais, M. Anthimopoulos, S. Shevchik, R. H. Botwey, D. Duke, C. Stettler, P. Diem, S. Mougiakakou *et al.*, "Carbohydrate estimation by a mobile phone-based system versus self-estimations of individuals with type 1 diabetes mellitus: a comparative study," *Journal of Medical Internet Research*, vol. 18, no. 5, p. e5567, 2016.

[52] Z. Huang, M. Soljak, B. O. Boehm, and J. Car, "Clinical relevance of smartphone apps for diabetes management: A global overview," *Diabetes/Metabolism Research and Reviews*, vol. 34, no. 4, p. e2990, 2018.

[53] O. AlShorman, B. AlShorman, M. Alkhassaweneh, and F. Alkahtani, "A review of internet of medical things (iomt)–based remote health monitoring through wearable sensors: A case study for diabetic patients," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 20, no. 1, pp. 414–422, 2020.

[54] A. Chakrabarty, S. Zavitsanou, T. Sowrirajan, F. J. Doyle III, and E. Dassau, "Getting IoT-ready: The face of next generation artificial pancreas systems," in *The Artificial Pancreas.* Elsevier, 2019, pp. 29–57.

[55] Z. A. Al-Odat, S. K. Srinivasan, E. M. Al-Qtiemat, and S. Shuja, "A reliable IoT-based embedded health care system for diabetic patients," *International Journal on Advances in Internet Technology Volume 12, Number 1 & 2, 2019*, 2019.

[56] P. Herrero, M. El-Sharkawy, J. Daniels, N. Jugnee, C. N. Uduku, M. Reddy, N. Oliver, and P. Georgiou, "The bio-inspired artificial pancreas for type 1 diabetes control in the home: System architecture and preliminary results," *Journal of Diabetes Science and Technology*, vol. 13, no. 6, pp. 1017–1025, 2019.

[57] P. Herrero, M. El Sharkawy, P. Pesl, M. Reddy, N. Oliver, D. Johnston, C. Toumazou, and P. Georgiou, "Live demonstration: A handheld bio-inspired artificial pancreas for treatment of diabetes," in *IEEE Biomedical Circuits and Systems Conference (BioCAS) Proceedings*. IEEE, 2014, pp. 172–172.

[58] M. D. Breton, D. R. Cherñavvsky, G. P. Forlenza, M. D. DeBoer, J. Robic, R. P. Wadwa, L. H. Messer, B. P. Kovatchev, and D. M. Maahs, "Closed-loop control during intense prolonged outdoor exercise in adolescents with type 1 diabetes: the artificial pancreas ski study," *Diabetes Care*, vol. 40, no. 12, pp. 1644–1650, 2017.

[59] P. V. Astillo, G. Choudhary, D. G. Duguma, J. Kim, and I. You, "Trmaps: Trust management in specification-based misbehavior detection system for imd-enabled artificial pancreas system," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 10, pp. 3763–3775, 2021.

[60] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[61] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[62] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[63] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, and Y. Bengio, "Theano: a CPU and GPU math ex-

pression compiler," in *Proceedings of the Python for scientific computing conference (SciPy)*, vol. 4, no. 3. Austin, TX, 2010.

[64] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*, 2014, pp. 675–678.

[65] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, 2016, pp. 265–283.

[66] F. Seide and A. Agarwal, "CNTK: Microsoft's open-source deep-learning toolkit," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 2135–2135.

[67] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, 2019, pp. 8024–8035.

[68] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[69] D. C. Cireşan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Deep, big, simple neural nets for handwritten digit recognition," *Neural Computation*, vol. 22, no. 12, pp. 3207–3220, 2010.

[70] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition.* IEEE, 2009, pp. 248–255.

[71] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.

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

[73] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder–decoder approaches," in *8th Workshop on Syntax, Semantics and Structure in Statistical Translation.* Association for Computational Linguistics, 2014, pp. 103–111.

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

[75] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1334–1373, 2016.

[76] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton *et al.*, "Mastering the game of go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[77] M. Popova, O. Isayev, and A. Tropsha, "Deep reinforcement learning for de novo drug design," *Science Advances*, vol. 4, no. 7, p. eaap7885, 2018.

[78] J. Degrave, F. Felici, J. Buchli, M. Neunert, B. Tracey, F. Carpanese, T. Ewalds, R. Hafner, A. Abdolmaleki, D. de las Casas, C. Donner, L. Fritz, C. Galperti, A. Huber, J. Keeling, M. Tsimpoukelli, J. Kay, A. Merle, J.-M. Moret, S. Noury, F. Pesamosca, D. Pfau, O. Sauter, C. Sommariva, S. Coda, B. Duval, A. Fasoli, P. Kohli, K. Kavukcuoglu, D. Hassabis, and M. Riedmiller, "Magnetic control of tokamak plasmas through deep reinforcement learning," *Nature*, vol. 602, no. 7897, pp. 414–419, 2022.

[79] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction.* MIT press, 2018.

[80] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[81] C. Dalla Man, F. Micheletto, D. Lv, M. Breton, B. Kovatchev, and C. Cobelli, "The UVA/PADOVA type 1 diabetes simulator: new features," *Journal of Diabetes Science and Technology*, vol. 8, no. 1, pp. 26–34, 2014.

[82] G. E. Hinton, "Deep belief networks," *Scholarpedia*, vol. 4, no. 5, p. 5947, 2009.

[83] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[84] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.

[85] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019.

[86] D. Moher, A. Liberati, J. Tetzlaff, and D. G. Altman, "Preferred reporting items for systematic reviews and meta-analyses: the PRISMA statement," *Annals of Internal Medicine*, vol. 151, no. 4, pp. 264–269, 2009.

[87] P. P. San, S. H. Ling, and H. T. Nguyen, "Deep learning framework for detection of hypoglycemic episodes in children with type 1 diabetes," in *38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC).* IEEE, 2016, pp. 3503–3506.

[88] S. Mirshekarian, R. Bunescu, C. Marling, and F. Schwartz, "Using LSTMs to learn physiological models of blood glucose behavior," in *39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC).* IEEE, 2017, pp. 2887–2891.

[89] Q. Sun, M. V. Jankovic, L. Bally, and S. G. Mougiakakou, "Predicting blood glucose with an LSTM and Bi-LSTM based deep neural network," in *14th Symposium on Neural Networks and Applications (NEUREL)*. IEEE, 2018, pp. 1–5.

[90] M. Anthimopoulos, J. Dehais, S. Shevchik, B. H. Ransford, D. Duke, P. Diem, and S. Mougiakakou, "Computer vision-based carbohydrate estimation for type 1 patients with diabetes using smartphones," *Journal of Diabetes Science and Technology*, vol. 9, no. 3, pp. 507–515, 2015.

[91] I. Fox, L. Ang, M. Jaiswal, R. Pop-Busui, and J. Wiens, "Deep multi-output forecasting: Learning to accurately predict blood glucose trajectories," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 1387–1395.

[92] A. Chakrabarty, F. J. Doyle, and E. Dassau, "Deep learning assisted macronutrient estimation for feedforward-feedback control in artificial pancreas systems," in *Annual American Control Conference (ACC)*. IEEE, 2018, pp. 3564–3570.

[93] L. Bossard, M. Guillaumin, and L. V. Gool, "Food-101–mining discriminative components with random forests," in *European Conference on Computer Vision*. Springer, 2014, pp. 446–461.

[94] S. H. A. Faruqui, Y. Du, R. Meka, A. Alaeddini, C. Li, S. Shirinkam, and J. Wang, "Development of a deep learning model for dynamic forecasting of blood glucose level for type 2 diabetes mellitus: Secondary analysis of a randomized controlled trial," *JMIR mHealth and uHealth*, vol. 7, no. 11, p. e14452, 2019.

[95] N. S. Padhye and J. Wang, "Pattern of active and inactive sequences of diabetes self-monitoring in mobile phone and paper diary users," in *37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2015, pp. 7630–7633.

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

[97] A. Aliberti, I. Pupillo, S. Terna, E. Macii, S. Di Cataldo, E. Patti, and A. Acquaviva, "A multi-patient data-driven approach to blood glucose prediction," *IEEE Access*, vol. 7, pp. 69 311–69 325, 2019.

[98] Jaeb Center for Health Research (JCHR), "Diabetes research studies," 2018. [Online]. Available: https://public.jaeb.org/datasets/diabetes

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

[100] C. Marling and R. Bunescu, "The OhioT1DM dataset for blood glucose level prediction: Update 2020," in *The 5th International Workshop on Knowledge Discovery in Healthcare Data in the 24th ECAI*, 2020, pp. 71–74.

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

[102] I. Fox and J. Wiens, "Reinforcement learning for blood glucose control: Challenges and opportunities," in *Reinforcement Learning for Real Life (RL4RealLife) Workshop in the 36th International Conference on Machine Learning*, 2019.

[103] B. P. Kovatchev, M. Breton, C. Dalla Man, and C. Cobelli, "In silico preclinical trials: A proof of concept in closed-loop control of type 1 diabetes," *Journal of Diabetes Science and Technology*, vol. 3, no. 1, pp. 44–55, 2009.

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

[105] C. C. Palerm, H. Zisser, L. Jovanovič, and F. J. Doyle III, "A run-to-run control strategy to adjust basal insulin infusion rates in type 1 diabetes," *Journal of Process Control*, vol. 18, no. 3-4, pp. 258–265, 2008.

[106] A. Zaitcev, M. R. Eissa, Z. Hui, T. Good, J. Elliott, and M. Benaissa, “A deep neural network application for improved prediction of HbA1c in type 1 diabetes,” *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 10, pp. 2932–2941, 2020.

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

[108] H. Hameed and S. Kleinberg, “Comparing machine learning techniques for blood glucose forecasting using free-living and patient generated data,” in *Machine Learning for Healthcare Conference.* PMLR, 2020, pp. 871–894.

[109] A. Melmer, T. Züger, D. M. Lewis, S. Leibrand, C. Stettler, and M. Laimer, “Glycaemic control in individuals with type 1 diabetes using an open source artificial pancreas system (OpenAPS),” *Diabetes, Obesity and Metabolism*, vol. 21, no. 10, pp. 2333–2337, 2019.

[110] D. N. Thyde, A. Mohebbi, H. Bengtsson, M. L. Jensen, and M. Mørup, “Machine learning-based adherence detection of type 2 diabetes patients on once-daily basal insulin injections,” *Journal of Diabetes Science and Technology*, vol. 15, no. 1, pp. 98–108, 2021.

[111] S. S. Kanderian, S. Weinzimer, G. Voskanyan, and G. M. Steil, “Identification of intraday metabolic profiles during closed-loop glucose control in individuals with type 1 diabetes,” *Journal of Diabetes Science and Technology*, vol. 3, no. 5, pp. 1047–1057, 2009.

[112] M. De Bois, M. A. El Yacoubi, and M. Ammi, “Adversarial multi-source transfer learning in healthcare: Application to glucose prediction for diabetic people,” *Computer Methods and Programs in Biomedicine*, vol. 199, p. 105874, 2021.

[113] J. Beauchamp, R. Bunescu, C. Marling, Z. Li, and C. Liu, “LSTMs and deep residual networks for carbohydrate and bolus recommendations in type 1 diabetes management,” *Sensors*, vol. 21, no. 9, p. 3303, 2021.

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

[115] X. Sun, Y. M. Bee, S. W. Lam, Z. Liu, W. Zhao, S. Y. Chia, H. A. Kadir, J. T. Wu, B. Y. Ang, N. Liu *et al.*, "Effective treatment recommendations for type 2 diabetes management using reinforcement learning: Treatment recommendation model development and validation," *Journal of Medical Internet Research*, vol. 23, no. 7, p. e27858, 2021.

[116] D. Y. Z. Lim, S. Y. Chia, H. A. Kadir, N. N. M. Salim, and Y. M. Bee, "Establishment of the singhealth diabetes registry," *Clinical Epidemiology*, vol. 13, p. 215, 2021.

[117] J. Daniels, P. Herrero, and P. Georgiou, "A multitask learning approach to personalized blood glucose prediction," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 1, pp. 436–445, 2021.

[118] W. Seo, S.-W. Park, N. Kim, S.-M. Jin, and S.-M. Park, "A personalized blood glucose level prediction model with a fine-tuning strategy: A proof-of-concept study," *Computer Methods and Programs in Biomedicine*, vol. 211, p. 106424, 2021.

[119] J. H. Yoo, M. S. Choi, J. Ahn, S. W. Park, Y. Kim, K. Y. Hur, S.-M. Jin, G. Kim, and J. H. Kim, "Association between continuous glucose monitoring-derived time in range, other core metrics, and albuminuria in type 2 diabetes," *Diabetes Technology & Therapeutics*, vol. 22, no. 10, pp. 768–776, 2020.

[120] S. M. A. Zaidi, V. Chandola, M. Ibrahim, B. Romanski, L. D. Mastrandrea, and T. Singh, "Multi-step ahead predictive model for blood glucose concentrations of type-1 diabetic patients," *Scientific Reports*, vol. 11, no. 1, pp. 1–14, 2021.

[121] Tidepool, "Make a big impact on diabetes research," 2018. [Online]. Available: https://www.tidepool.org/

[122] H. Nemat, H. Khadem, M. R. Eissa, J. Elliott, and M. Benaissa, “Blood glucose level prediction: Advanced deep-ensemble learning approach,” *IEEE Journal of Biomedical and Health Informatics*, pp. 1–1, 2022.

[123] J. Noguer, I. Contreras, O. Mujahid, A. Beneyto, and J. Vehi, “Generation of individualized synthetic data for augmentation of the type 1 diabetes data sets using deep learning models,” *Sensors*, vol. 22, no. 13, p. 4944, 2022.

[124] A. Bertachi, C. Viñals, L. Biagi, I. Contreras, J. Vehí, I. Conget, and M. Giménez, “Prediction of nocturnal hypoglycemia in adults with type 1 diabetes under multiple daily injections using continuous glucose monitoring and physical activity monitor,” *Sensors*, vol. 20, no. 6, p. 1705, 2020.

[125] P. Viroonluecha, E. Egea-Lopez, and J. Santa, “Evaluation of blood glucose level control in type 1 diabetic patients using deep reinforcement learning,” *PLOS One*, vol. 17, no. 9, p. e0274608, 2022.

[126] T. Zhu, K. Li, P. Herrero, J. Chen, and P. Georgiou, “A deep learning algorithm for personalized blood glucose prediction,” in *The 3rd International Workshop on Knowledge Discovery in Healthcare Data in the 27th IJCAI-ECAI*, 2018, pp. 74–78.

[127] J. Chen, K. Li, P. Herrero, T. Zhu, and P. Georgiou, “Dilated recurrent neural network for short-time prediction of glucose concentration.” in *The 3rd International Workshop on Knowledge Discovery in Healthcare Data in the 27th IJCAI-ECAI*, 2018, pp. 69–73.

[128] T. Zhu, K. Li, J. Chen, P. Herrero, and P. Georgiou, “Dilated recurrent neural networks for glucose forecasting in type 1 diabetes,” *Journal of Healthcare Informatics Research*, vol. 4, no. 3, pp. 308–324, 2020.

[129] M. Vettoretti, S. Del Favero, G. Sparacino, and A. Facchinetti, “Modeling the error of factory-calibrated continuous glucose monitoring sensors: application to dexcom g6 sensor data,” in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2019, pp. 750–753.

[130] A. Brazeau, H. Mircescu, K. Desjardins, C. Leroux, I. Strychar, J. Ekoé, and R. Rabasa-Lhoret, "Carbohydrate counting accuracy and blood glucose variability in adults with type 1 diabetes," *Diabetes Research and Clinical Practice*, vol. 99, no. 1, pp. 19–23, 2013.

[131] M. Porumb, S. Stranges, A. Pescapè, and L. Pecchia, "Precision medicine and artificial intelligence: A pilot study on deep learning for hypoglycemic events detection based on ECG," *Scientific Reports*, vol. 10, no. 1, pp. 1–16, 2020.

[132] T. J. Loftus, B. Shickel, M. M. Ruppert, J. A. Balch, T. Ozrazgat-Baslanti, P. J. Tighe, P. A. Efron, W. R. Hogan, P. Rashidi, G. R. Upchurch Jr *et al.*, "Uncertainty-aware deep learning in healthcare: A scoping review," *PLOS Digital Health*, vol. 1, no. 8, p. e0000085, 2022.

[133] S. M. Lundberg, B. Nair, M. S. Vavilala, M. Horibe, M. J. Eisses, T. Adams, D. E. Liston, D. K.-W. Low, S.-F. Newman, J. Kim *et al.*, "Explainable machine-learning predictions for the prevention of hypoxaemia during surgery," *Nature Biomedical Engineering*, vol. 2, no. 10, pp. 749–760, 2018.

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

[135] G. Sparacino, F. Zanderigo, S. Corazza, A. Maran, A. Facchinetti, and C. Cobelli, "Glucose concentration can be predicted ahead in time from continuous glucose monitoring sensor time-series," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 5, pp. 931–937, 2007.

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

[137] E. Montaser, J.-L. Díez, P. Rossetti, M. Rashid, A. Cinar, and J. Bondia, "Seasonal local models for glucose prediction in type 1 diabetes," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 7, pp. 2064–2072, 2019.

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

[139] J. E. Pinsker, A. J. Laguna Sanz, J. B. Lee, M. M. Church, C. Andre, L. E. Lindsey, F. J. Doyle III, and E. Dassau, "Evaluation of an artificial pancreas with enhanced model predictive control and a glucose prediction trust index with unannounced exercise," *Diabetes Technology & Therapeutics*, vol. 20, no. 7, pp. 455–464, 2018.

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

[141] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations*, 2015, pp. 1–15.

[142] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International Conference on Machine Learning*. PMLR, 2015, pp. 2048–2057.

[143] A. Galassi, M. Lippi, and P. Torroni, "Attention in natural language processing," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 10, pp. 4291–4308, 2021.

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

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

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

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

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

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

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

[151] T. Zhu, X. Yao, K. Li, P. Herrero, and P. Georgiou, "Blood glucose prediction for type 1 diabetes using generative adversarial networks," in *The 5th International Workshop on Knowledge Discovery in Healthcare Data in the 24th ECAI*, 2020, pp. 90–94.

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

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

[154] A. Nichol, J. Achiam, and J. Schulman, "On first-order meta-learning algorithms," *CoRR*, vol. abs/1803.02999, 2018. [Online]. Available: http://arxiv.org/abs/1803.02999

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

[156] P. Calhoun, T. K. Johnson, J. Hughes, D. Price, and A. K. Balo, "Resistance to acetaminophen interference in a novel continuous glucose monitoring system," *Journal of Diabetes Science and Technology*, vol. 12, no. 2, pp. 393–396, 2018.

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

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

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

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

[161] K. Plis, R. Bunescu, C. Marling, J. Shubrook, and F. Schwartz, "A machine learning approach to predicting blood glucose levels for diabetes management," in *Workshops at the 28th AAAI conference on artificial intelligence*, 2014.

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

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

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

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

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

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

[168] N. Hobbs, I. Hajizadeh, M. Rashid, K. Turksoy, M. Breton, and A. Cinar, "Improving glucose prediction accuracy in physically active adolescents with type 1 diabetes," *Journal of Diabetes Science and Technology*, vol. 13, no. 4, pp. 718–727, 2019.

[169] W. P. van Doorn, Y. D. Foreman, N. C. Schaper, H. H. Savelberg, A. Koster, C. J. van der Kallen, A. Wesselius, M. T. Schram, R. M. Henry, P. C. Dagnelie *et al.*, "Machine learning-based glucose prediction with use of continuous glucose and physical activity monitoring data: The Maastricht study," *PLOS One*, vol. 16, no. 6, p. e0253125, 2021.

[170] M. Sevil, M. Rashid, Z. Maloney, I. Hajizadeh, S. Samadi, M. R. Askari, N. Hobbs, R. Brandt, M. Park, L. Quinn *et al.*, "Determining physical activity characteristics from wristband data for use in automated insulin delivery systems," *IEEE Sensors Journal*, vol. 20, no. 21, pp. 12 859–12 870, 2020.

[171] B. Ozaslan, S. D. Patek, and M. D. Breton, "Impact of daily physical activity as measured by commonly available wearables on mealtime glucose control in type 1 diabetes," *Diabetes Technology & Therapeutics*, vol. 22, no. 10, pp. 742–748, 2020.

[172] B. Bent, P. J. Cho, M. Henriquez, A. Wittmann, C. Thacker, M. Feinglos, M. J. Crowley, and J. P. Dunn, "Engineering digital biomarkers of interstitial glucose from noninvasive smartwatches," *npj Digital Medicine*, vol. 4, no. 1, pp. 1–11, 2021.

[173] Y. Wu, X. Yao, G. Vespasiani, A. Nicolucci, Y. Dong, J. Kwong, L. Li, X. Sun, H. Tian, and S. Li, "Mobile app-based interventions to support diabetes self-management: a systematic review of randomized controlled trials to identify functions associated with glycemic efficacy," *JMIR mHealth and uHealth*, vol. 5, no. 3, p. e35, 2017.

[174] C. Pérez-Gandía, A. Facchinetti, G. Sparacino, C. Cobelli, E. Gómez, M. Rigla, A. de Leiva, and M. Hernando, "Artificial neural network algorithm for online glucose prediction from continuous glucose monitoring," *Diabetes Technology & Therapeutics*, vol. 12, no. 1, pp. 81–88, 2010.

[175] K. Larsen, J. H. Petersen, E. Budtz-Jørgensen, and L. Endahl, "Interpreting parameters in the logistic regression model with random effects," *Biometrics*, vol. 56, no. 3, pp. 909–914, 2000.

[176] E. Peper, R. Harvey, I.-M. Lin, H. Tylova, and D. Moss, “Is there more to blood volume pulse than heart rate variability, respiratory sinus arrhythmia, and cardiorespiratory synchrony?” *Biofeedback*, vol. 35, no. 2, 2007.

[177] W. Zong, T. Heldt, G. Moody, and R. Mark, “An open-source algorithm to detect onset of arterial blood pressure pulses,” in *Computers in Cardiology.* IEEE, 2003, pp. 259–262.

[178] M. O. Bekkink, M. Koeneman, B. E. de Galan, and S. J. Bredie, “Early detection of hypoglycemia in type 1 diabetes using heart rate variability measured by a wearable device,” *Diabetes Care*, vol. 42, no. 4, pp. 689–692, 2019.

[179] Electrophysiology, Task Force of the European Society of Cardiology the North American Society of Pacing, “Heart rate variability: standards of measurement, physiological interpretation, and clinical use,” *Circulation*, vol. 93, no. 5, pp. 1043–1065, 1996.

[180] M. Benedek and C. Kaernbach, “A continuous measure of phasic electrodermal activity,” *Journal of Neuroscience Methods*, vol. 190, no. 1, pp. 80–91, 2010.

[181] C. Carreiras, A. P. Alves, A. Lourenço, F. Canento, H. Silva, A. Fred *et al.*, “BioSPPy: Biosignal processing in Python,” 2015. [Online]. Available: https://github.com/PIA-Group/BioSPPy/

[182] D. Makowski, T. Pham, Z. J. Lau, J. C. Brammer, F. Lespinasse, H. Pham, C. Schölzel, and A. S H Chen, “Neurokit2: A Python toolbox for neurophysiological signal processing,” 2020. [Online]. Available: https://github.com/neuropsychology/NeuroKit

[183] S. L. Cichosz, J. Frystyk, O. K. Hejlesen, L. Tarnow, and J. Fleischer, “A novel algorithm for prediction and detection of hypoglycemia based on continuous glucose monitoring and heart rate variability in patients with type 1 diabetes,” *Journal of Diabetes Science and Technology*, vol. 8, no. 4, pp. 731–737, 2014.

[184] C. Marling, L. Xia, R. Bunescu, and F. Schwartz, “Machine learning experiments with noninvasive sensors for hypoglycemia detection,” in *IJCAI Workshop on Knowledge Discovery in Healthcare Data*, 2016, pp. 1–6.

[185] D. R. Roberts, V. Bahn, S. Ciuti, M. S. Boyce, J. Elith, G. Guillera-Arroita, S. Hauenstein, J. J. Lahoz-Monfort, B. Schröder, W. Thuiller *et al.*, “Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure,” *Ecography*, vol. 40, no. 8, pp. 913–929, 2017.

[186] K. Turksoy, E. S. Bayrak, L. Quinn, E. Littlejohn, D. Rollins, and A. Cinar, “Hypoglycemia early alarm systems based on multivariable models,” *Industrial & Engineering Chemistry Research*, vol. 52, no. 35, pp. 12 329–12 336, 2013.

[187] P. Herrero, R. Calm, J. Vehí, J. Armengol, P. Georgiou, N. Oliver, and C. Tomazou, “Robust fault detection system for insulin pump therapy using continuous glucose monitoring,” *Journal of Diabetes Science and Technology*, vol. 6, no. 5, pp. 1131–1141, 2012.

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

[189] C. Liu, P. Avari, Y. Leal, M. Wos, K. Sivasithamparam, P. Georgiou, M. Reddy, J. M. Fernández-Real, C. Martin, M. Fernández-Balsells *et al.*, “A modular safety system for an insulin dose recommender: a feasibility study,” *Journal of Diabetes Science and Technology*, vol. 14, no. 1, pp. 87–96, 2020.

[190] F. J. Doyle, L. M. Huyett, J. B. Lee, H. C. Zisser, and E. Dassau, “Closed-loop artificial pancreas systems: engineering the algorithms,” *Diabetes Care*, vol. 37, no. 5, pp. 1191–1197, 2014.

[191] R. Hovorka, “Artificial pancreas project at Cambridge 2013,” *Diabetic Medicine*, vol. 32, no. 8, pp. 987–992, 2015.

[192] G. P. Forlenza, S. Deshpande, T. T. Ly, D. P. Howsmon, F. Cameron, N. Baysal, E. Mauritzen, T. Marcal, L. Towers, B. W. Bequette *et al.*, "Application of zone model predictive control artificial pancreas during extended use of infusion set and sensor: a randomized crossover-controlled home-use trial," *Diabetes Care*, vol. 40, no. 8, pp. 1096–1102, 2017.

[193] L. H. Messer, G. P. Forlenza, J. L. Sherr, R. P. Wadwa, B. A. Buckingham, S. A. Weinzimer, D. M. Maahs, and R. H. Slover, "Optimizing hybrid closed-loop therapy in adolescents and emerging adults using the MiniMed 670G system," *Diabetes Care*, vol. 41, no. 4, pp. 789–796, 2018.

[194] E. Atlas, R. Nimri, S. Miller, E. A. Grunberg, and M. Phillip, "MD-logic artificial pancreas system: a pilot study in adults with type 1 diabetes," *Diabetes Care*, vol. 33, no. 5, pp. 1072–1076, 2010.

[195] L. E. Castellanos, C. A. Balliro, J. S. Sherwood, R. Jafri, M. A. Hillard, E. Greaux, R. Selagamsetty, H. Zheng, F. H. El-Khatib, E. R. Damiano *et al.*, "Performance of the insulin-only ilet bionic pancreas and the bihormonal ilet using dasiglucagon in adults with type 1 diabetes in a home-use setting," *Diabetes Care*, vol. 44, no. 6, pp. e118–e120, 2021.

[196] H. Blauw, A. J. Onvlee, M. Klaassen, A. C. van Bon, and J. H. DeVries, "Fully closed loop glucose control with a bihormonal artificial pancreas in adults with type 1 diabetes: an outpatient, randomized, crossover trial," *Diabetes Care*, vol. 44, no. 3, pp. 836–838, 2021.

[197] C. Owens, H. Zisser, L. Jovanovic, B. Srinivasan, D. Bonvin, and F. J. Doyle, "Run-to-run control of blood glucose concentrations for people with type 1 diabetes mellitus," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 6, pp. 996–1005, 2006.

[198] C. C. Palerm, H. Zisser, W. C. Bevier, L. Jovanovič, and F. J. Doyle, "Prandial insulin dosing using run-to-run control: application of clinical data and medical

expertise to define a suitable performance metric," *Diabetes Care*, vol. 30, no. 5, pp. 1131–1136, 2007.

[199] P. Pesl, P. Herrero, M. Reddy, M. Xenou, N. Oliver, D. Johnston, C. Toumazou, and P. Georgiou, "An advanced bolus calculator for type 1 diabetes: system architecture and usability results," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 1, pp. 11–17, 2015.

[200] M. Reddy, P. Pesl, M. Xenou, C. Toumazou, D. Johnston, P. Georgiou, P. Herrero, and N. Oliver, "Clinical safety and feasibility of the advanced bolus calculator for type 1 diabetes based on case-based reasoning: a 6-week nonrandomized single-arm pilot study," *Diabetes Technology & Therapeutics*, vol. 18, no. 8, pp. 487–493, 2016.

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

[202] N. S. Tyler, C. M. Mosquera-Lopez, L. M. Wilson, R. H. Dodier, D. L. Branigan, V. B. Gabo, F. H. Guillot, W. W. Hilts, J. El Youssef, J. R. Castle *et al.*, "An artificial intelligence decision support system for the management of type 1 diabetes," *Nature Metabolism*, pp. 1–8, 2020.

[203] M. Lee, T. M. Gatton, and K.-K. Lee, "A monitoring and advisory system for diabetes patient management using a rule-based method and KNN," *Sensors*, vol. 10, no. 4, pp. 3934–3953, 2010.

[204] E. M. Aiello, C. Toffanin, M. Messori, C. Cobelli, and L. Magni, "Postprandial glucose regulation via knn meal classification in type 1 diabetes," *IEEE Control Systems Letters*, vol. 3, no. 2, pp. 230–235, 2018.

[205] G. Cappon, M. Vettoretti, F. Marturano, A. Facchinetti, and G. Sparacino, "A neural-network-based approach to personalize insulin bolus calculation using continuous glucose monitoring," *Journal of Diabetes Science and Technology*, vol. 12, no. 2, pp. 265–272, 2018.

[206] Q. Sun, M. V. Jankovic, J. Budzinski, B. Moore, P. Diem, C. Stettler, and S. G. Mougiakakou, "A dual mode adaptive basal-bolus advisor based on reinforcement learning," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 6, pp. 2633–2641, 2018.

[207] A. Agianniotis, M. Anthimopoulos, E. Daskalaki, A. Drapela, C. Stettler, P. Diem, and S. Mougiakakou, "GoCARB in the context of an artificial pancreas," *Journal of Diabetes Science and Technology*, vol. 9, no. 3, pp. 549–555, 2015.

[208] G. P. Forlenza, L. Ekhlaspour, M. Breton, D. M. Maahs, R. P. Wadwa, M. DeBoer, L. H. Messer, M. Town, J. Pinnata, G. Kruse *et al.*, "Successful at-home use of the Tandem Control-IQ artificial pancreas system in young children during a randomized controlled trial," *Diabetes Technology & Therapeutics*, vol. 21, no. 4, pp. 159–169, 2019.

[209] M. Tejedor, A. Z. Woldaregay, and F. Godtliebsen, "Reinforcement learning application in diabetes blood glucose control: A systematic review," *Artificial Intelligence in Medicine*, p. 101836, 2020.

[210] W. J. Artman, I. Nahum-Shani, T. Wu, J. R. Mckay, and A. Ertefaie, "Power analysis in a SMART design: sample size estimation for determining the best embedded dynamic treatment regime," *Biostatistics*, 2018.

[211] R. Visentin, C. Dalla Man, B. Kovatchev, and C. Cobelli, "The university of Virginia/Padova type 1 diabetes simulator matches the glucose traces of a clinical trial," *Diabetes Technology & Therapeutics*, vol. 16, no. 7, pp. 428–434, 2014.

[212] S. Lee, J. Kim, S. W. Park, S.-M. Jin, and S.-M. Park, "Toward a fully automated artificial pancreas system using a bioinspired reinforcement learning design: In silico validation," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 2, pp. 536–546, 2020.

[213] I. Fox, J. Lee, R. Pop-Busui, and J. Wiens, "Deep reinforcement learning for closed-loop blood glucose control," in *Machine Learning for Healthcare Conference.* PMLR, 2020, pp. 508–536.

[214] J. Nordhaug Myhre, M. Tejedor, I. Kalervo Launonen, A. El Fathi, and F. Godtliebsen, "In-silico evaluation of glucose regulation using policy gradient reinforcement learning for patients with type 1 diabetes mellitus," *Applied Sciences*, vol. 10, no. 18, p. 6350, 2020.

[215] R. Hovorka, V. Canonico, L. J. Chassin, U. Haueter, M. Massi-Benedetti, M. O. Federici, T. R. Pieber, H. C. Schaller, L. Schaupp, T. Vering *et al.*, "Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes," *Physiological Measurement*, vol. 25, no. 4, p. 905, 2004.

[216] P. Herrero, P. Pesl, J. Bondia, M. Reddy, N. Oliver, P. Georgiou, and C. Toumazou, "Method for automatic adjustment of an insulin bolus calculator: in silico robustness evaluation under intra-day variability," *Computer Methods and Programs in Biomedicine*, vol. 119, no. 1, pp. 1–8, 2015.

[217] S. Fujimoto, D. Meger, and D. Precup, "Off-policy deep reinforcement learning without exploration," in *International Conference on Machine Learning.* PMLR, 2019, pp. 2052–2062.

[218] S. Fujimoto and S. S. Gu, "A minimalist approach to offline reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[219] A. Kumar, J. Fu, M. Soh, G. Tucker, and S. Levine, "Stabilizing off-policy q-learning via bootstrapping error reduction," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[220] A. Kumar, A. Zhou, G. Tucker, and S. Levine, "Conservative q-learning for offline reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1179–1191, 2020.

[221] I. Kostrikov, R. Fergus, J. Tompson, and O. Nachum, "Offline reinforcement learning with fisher divergence critic regularization," in *International Conference on Machine Learning.* PMLR, 2021, pp. 5774–5783.

[222] Y. Xie, B. Liu, Q. Liu, Z. Wang, Y. Zhou, and J. Peng, "Off-policy evaluation and learning from logged bandit feedback: Error reduction via surrogate policy," in *International Conference on Learning Representations*, 2018.

[223] H. Le, C. Voloshin, and Y. Yue, "Batch policy learning under constraints," in *International Conference on Machine Learning*. PMLR, 2019, pp. 3703–3712.

[224] M. R. Zhang, T. Paine, O. Nachum, C. Paduraru, G. Tucker, M. Norouzi *et al.*, "Autoregressive dynamics models for offline policy evaluation and optimization," in *International Conference on Learning Representations*, 2021.

[225] T. L. Paine, C. Paduraru, A. Michi, C. Gulcehre, K. Zolna, A. Novikov, Z. Wang, and N. de Freitas, "Hyperparameter selection for offline reinforcement learning," in *The Offline Reinforcement Learning Workshop, NeurIPS 2020*, 2020.

[226] J. Fu, M. Norouzi, O. Nachum, G. Tucker, A. Novikov, M. Yang, M. R. Zhang, Y. Chen, A. Kumar, C. Paduraru *et al.*, "Benchmarks for deep off-policy evaluation," in *International Conference on Learning Representations*, 2021.

[227] S. Tang and J. Wiens, "Model selection for offline reinforcement learning: Practical considerations for healthcare settings," in *Machine Learning for Healthcare Conference*. PMLR, 2021, pp. 2–35.

[228] S. Schmidt and K. Nørgaard, "Bolus calculators," *Journal of Diabetes Science and Technology*, vol. 8, no. 5, pp. 1035–1041, 2014.

[229] B. Kovatchev, "A century of diabetes technology: signals, models, and artificial pancreas control," *Trends in Endocrinology & Metabolism*, 2019.

[230] P. Herrero, J. Bondia, N. Oliver, and P. Georgiou, "A coordinated control strategy for insulin and glucagon delivery in type 1 diabetes," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 20, no. 13, pp. 1474–1482, 2017.

[231] J. R. Castle, J. E. Youssef, D. Branigan, B. Newswanger, P. Strange, M. Cummins, L. Shi, and S. Prestrelski, "Comparative pharmacokinetic/pharmacodynamic study

of liquid stable glucagon versus lyophilized glucagon in type 1 diabetes subjects," *Journal of Diabetes Science and Technology*, vol. 10, no. 5, pp. 1101–1107, 2016.

[232] I. Sala-Mira, P. Garcia, J.-L. Díez, and J. Bondia, "Internal model control based module for the elimination of meal and exercise announcements in hybrid artificial pancreas systems," *Computer Methods and Programs in Biomedicine*, p. 107061, 2022.

[233] A. Revert, F. Garelli, J. Picó, H. De Battista, P. Rossetti, J. Vehí, and J. Bondia, "Safety auxiliary feedback element for the artificial pancreas in type 1 diabetes," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 8, pp. 2113–2122, 2013.

[234] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," in *4th International Conference on Learning Representations*, 2016.

[235] T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, I. Osband *et al.*, "Deep Q-learning from demonstrations," in *32th AAAI Conference on Artificial Intelligence*, 2018.

[236] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *30th AAAI Conference on Artificial Intelligence*, 2016.

[237] Y. Gao, H. Xu, J. Lin, F. Yu, S. Levine, and T. Darrell, "Reinforcement learning from imperfect demonstrations," in *35th International Conference on Machine Learning*, 2018.

[238] G. P. Forlenza, Z. Li, B. A. Buckingham, J. E. Pinsker, E. Cengiz, R. P. Wadwa, L. Ekhlaspour, M. M. Church, S. A. Weinzimer, E. Jost *et al.*, "Predictive low-glucose suspend reduces hypoglycemia in adults, adolescents, and children with type 1 diabetes in an at-home randomized crossover study: results of the prolog trial," *Diabetes Care*, vol. 41, no. 10, pp. 2155–2161, 2018.

[239] D. M. Maahs, B. A. Buckingham, J. R. Castle, A. Cinar, E. R. Damiano, E. Dassau, J. H. DeVries, F. J. Doyle, S. C. Griffen, A. Haidar *et al.*, "Outcome measures for

artificial pancreas clinical trials: a consensus report," *Diabetes Care*, vol. 39, no. 7, pp. 1175–1179, 2016.

[240] L. Magni, D. M. Raimondo, C. Dalla Man, M. Breton, S. Patek, G. De Nicolao, C. Cobelli, and B. P. Kovatchev, "Evaluating the efficacy of closed-loop glucose regulation via control-variability grid analysis," *Journal of Diabetes Science and Technology*, vol. 2, no. 4, pp. 630–635, 2008.

[241] P. D. Ngo, S. Wei, A. Holubová, J. Muzik, and F. Godtliebsen, "Control of blood glucose for type-1 diabetes by using reinforcement learning with feedforward algorithm," *Computational and Mathematical Methods in Medicine*, vol. 2018, 2018.

[242] A. Bertachi, C. M. Ramkissoon, J. Bondia, and J. Vehí, "Automated blood glucose control in type 1 diabetes: A review of progress and challenges," *Endocrinología, Diabetes y Nutrición (English ed.)*, vol. 65, no. 3, pp. 172–181, 2018.

[243] R. Mauseth, I. B. Hirsch, J. Bollyky, R. Kircher, D. Matheson, S. Sanda, and C. Greenbaum, "Use of a "fuzzy logic" controller in a closed-loop artificial pancreas," *Diabetes Technology & Therapeutics*, vol. 15, no. 8, pp. 628–633, 2013.

[244] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," in *4th International Conference on Learning Representations*, 2016.

[245] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *31st International Conference on Machine Learning*. JMLR.org, 2014, p. I–387–I–395.

[246] H. Zisser, L. Robinson, W. Bevier, E. Dassau, C. Ellingsen, F. J. Doyle III, and L. Jovanovic, "Bolus calculator: a review of four "smart" insulin pumps," *Diabetes Technology & Therapeutics*, vol. 10, no. 6, pp. 441–444, 2008.

[247] A. Bertachi, L. Biagi, A. Beneyto, and J. Vehí, "Dynamic rule-based algorithm to tune insulin-on-board constraints for a hybrid artificial pancreas system," *Journal of Healthcare Engineering*, vol. 2020, 2020.

[248] A. M. Al-Taee, M. A. Al-Taee, W. Al-Nuaimy, Z. J. Muhsin, and H. AlZu'bi, "Smart bolus estimation taking into account the amount of insulin on board," in *IEEE International Conference on Computer and Information Technology*, 2015, pp. 1051–1056.

[249] C. Toffanin, H. Zisser, F. J. Doyle III, and E. Dassau, "Dynamic insulin on board: incorporation of circadian insulin sensitivity variation," *Journal of Diabetes Science and Technology*, vol. 7, no. 4, pp. 928–940, 2013.

[250] P. Herrero, P. Pesl, M. Reddy, N. Oliver, P. Georgiou, and C. Toumazou, "Advanced insulin bolus advisor based on run-to-run control and case-based reasoning," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 3, pp. 1087–1096, 2014.

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

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

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

[254] T. Battelino, T. Danne, R. M. Bergenstal, S. A. Amiel, R. Beck, T. Biester, E. Bosi, B. A. Buckingham, W. T. Cefalu, K. L. Close *et al.*, "Clinical targets for continuous glucose monitoring data interpretation: recommendations from the international consensus on time in range," *Diabetes Care*, vol. 42, no. 8, pp. 1593–1603, 2019.

[255] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *International Conference on Machine Learning.* PMLR, 2018, pp. 1587–1596.

[256] T. Danne, O. Kordonouri, M. Holder, H. Haberland, S. Golembowski, K. Remus, S. Bläsig, T. Wadien, S. Zierow, R. Hartmann *et al.*, “Prevention of hypoglycemia by using low glucose suspend function in sensor-augmented pump therapy,” *Diabetes Technology & Therapeutics*, vol. 13, no. 11, pp. 1129–1134, 2011.

[257] M. I. Takuma Seno, “d3rlpy: An offline deep reinforcement library,” in *The Offline Reinforcement Learning Workshop, NeurIPS*, 2021.

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

[259] G. Cappon, A. Facchinetti, G. Sparacino, and S. Del Favero, “A bayesian framework to identify type 1 diabetes physiological models using easily accessible patient data,” in *41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC).* IEEE, 2019, pp. 6914–6917.

[260] G. Noaro, G. Cappon, M. Vettoretti, G. Sparacino, S. Del Favero, and A. Facchinetti, “Machine-learning based model to improve insulin bolus calculation in type 1 diabetes therapy,” *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 1, pp. 247–255, 2020.

[261] J. Kesavadev, S. Srinivasan, B. Saboo, M. Krishna B, G. Krishnan *et al.*, “The do-it-yourself artificial pancreas: a comprehensive review,” *Diabetes Therapy*, vol. 11, no. 6, pp. 1217–1235, 2020.

[262] R. Armiger, M. Reddy, N. S. Oliver, P. Georgiou, and P. Herrero, “An in silico head-to-head comparison of the do-it-yourself artificial pancreas loop and bio-inspired artificial pancreas control algorithms,” *Journal of Diabetes Science and Technology*, vol. 16, no. 1, pp. 29–39, 2022.

[263] N. Resalat, J. El Youssef, N. Tyler, J. Castle, and P. G. Jacobs, “A statistical virtual patient population for the glucoregulatory system in type 1 diabetes with integrated exercise model,” *PLOS One*, vol. 14, no. 7, p. e0217301, 2019.

[264] P. Herrero, J. Bondia, O. Adewuyi, P. Pesl, M. El-Sharkawy, M. Reddy, C. Toumazou, N. Oliver, and P. Georgiou, “Enhancing automatic closed-loop glucose control in type 1 diabetes with an adaptive meal bolus calculator–in silico evaluation under intra-day variability,” *Computer Methods and Programs in Biomedicine*, vol. 146, pp. 125–131, 2017.

[265] R. Visentin, C. Dalla Man, and C. Cobelli, “One-day bayesian cloning of type 1 diabetes subjects: toward a single-day uva/padova type 1 diabetes simulator,” *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 11, pp. 2416–2424, 2016.

[266] S. Boschert and R. Rosen, “Digital twin—the simulation aspect,” in *Mechatronic Futures*. Springer, 2016, pp. 59–74.

[267] H. Elayan, M. Aloqaily, and M. Guizani, “Digital twin for intelligent context-aware IoT healthcare systems,” *IEEE Internet of Things Journal*, 2021.

[268] O. Mogren, “C-RNN-GAN: A continuous recurrent neural network with adversarial training,” in *Constructive Machine Learning Workshop, NeurIPS*, 2016, p. 1.

[269] C. Esteban, S. L. Hyland, and G. Rätsch, “Real-valued (medical) time series generation with recurrent conditional GANs,” *CoRR*, vol. abs/1706.02633, 2017. [Online]. Available: http://arxiv.org/abs/1706.02633

[270] C. Donahue, J. McAuley, and M. Puckette, “Adversarial audio synthesis,” in *International Conference on Learning Representations*, 2018.

[271] J. Yoon, D. Jarrett, and M. van der Schaar, “Time-series generative adversarial networks,” *Advances in Neural Information Processing Systems*, vol. 32, pp. 5508–5518, 2019.

[272] T. Siegmund, L. Heinemann, R. Kolassa, and A. Thomas, “Discrepancies between blood glucose and interstitial glucose—technological artifacts or physiology: implications for selection of the appropriate therapeutic target,” *Journal of Diabetes Science and Technology*, vol. 11, no. 4, pp. 766–772, 2017.

[273] F. Huszár, "How (not) to train your generative model: Scheduled sampling, likelihood, adversary?" *CoRR*, vol. abs/1511.05101, 2015. [Online]. Available: http://arxiv.org/abs/1511.05101

[274] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," *Advances in Neural Information Processing Systems*, vol. 29, pp. 2234–2242, 2016.

[275] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4, pp. 433–459, 2010.

[276] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. 11, 2008.

[277] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of Wasserstein GANs," in *Advances in Neural Information Processing Systems*, 2017.

[278] M. Swan, "Sensor mania! the internet of things, wearable computing, objective metrics, and the quantified self 2.0," *Journal of Sensor and Actuator Networks*, vol. 1, no. 3, pp. 217–253, 2012.

[279] H. M. Rooijackers, E. C. Wiegers, M. van der Graaf, D. H. Thijssen, R. P. Kessels, C. J. Tack, and B. E. de Galan, "A single bout of high-intensity interval training reduces awareness of subsequent hypoglycemia in patients with type 1 diabetes," *Diabetes*, vol. 66, no. 7, pp. 1990–1998, 2017.

[280] A. Trifan, M. Oliveira, and J. L. Oliveira, "Passive sensing of health outcomes through smartphones: systematic review of current solutions and possible limitations," *JMIR mHealth and uHealth*, vol. 7, no. 8, p. e12649, 2019.

[281] W. Shi and S. Dustdar, "The promise of edge computing," *Computer*, vol. 49, no. 5, pp. 78–81, 2016.

[282] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[283] X. Xu, Y. Ding, S. X. Hu, M. Niemier, J. Cong, Y. Hu, and Y. Shi, "Scaling for edge inference of deep neural networks," *Nature Electronics*, vol. 1, no. 4, pp. 216–222, 2018.

[284] L. Lai, N. Suda, and V. Chandra, "CMSIS-NN: Efficient neural network kernels for arm cortex-m cpus," *CoRR*, vol. abs/1801.06601, 2018. [Online]. Available: http://arxiv.org/abs/1801.06601

[285] R. Spence and M. Apperley, "Data base navigation: an office environment for the professional," *Behaviour & Information Technology*, vol. 1, no. 1, pp. 43–54, 1982.

[286] A. Mohebbi, A. R. Johansen, N. Hansen, P. E. Christensen, J. M. Tarp, M. L. Jensen, H. Bengtsson, and M. Mørup, "Short term blood glucose prediction based on continuous glucose monitoring data," in *42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC).* IEEE, 2020, pp. 5140–5145.

[287] T. Danne, R. Nimri, T. Battelino, R. M. Bergenstal, K. L. Close, J. H. DeVries, S. Garg, L. Heinemann, I. Hirsch, S. A. Amiel *et al.*, "International consensus on use of continuous glucose monitoring," *Diabetes Care*, vol. 40, no. 12, pp. 1631–1640, 2017.

[288] D. Chicco and G. Jurman, "The advantages of the matthews correlation coefficient (MCC) over f1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 1, pp. 1–13, 2020.

[289] S. Bai, J. Z. Kolter, and V. Koltun, "Convolutional sequence modeling revisited," in *Workshop Track - Sixth International Conference on Learning Representations*, 2018.

[290] B. Lim, S. Ö. Arık, N. Loeff, and T. Pfister, "Temporal fusion transformers for interpretable multi-horizon time series forecasting," *International Journal of Forecasting*, vol. 37, no. 4, pp. 1748–1764, 2021.

[291] A. A. Schuurmans, P. de Looff, K. S. Nijhof, C. Rosada, R. H. Scholte, A. Popma, and R. Otten, "Validity of the empatica e4 wristband to measure heart rate variability (HRV) parameters: a comparison to electrocardiography (ECG)," *Journal of Medical Systems*, vol. 44, no. 11, pp. 1–11, 2020.

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

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

[294] M. Schiavon, C. Dalla Man, Y. C. Kudva, A. Basu, and C. Cobelli, "In silico optimization of basal insulin infusion rate during exercise: implication for artificial pancreas," *Journal of Diabetes Science and Technology*, vol. 7, no. 6, pp. 1461–1469, 2013.

[295] V. H. Pong, A. V. Nair, L. M. Smith, C. Huang, and S. Levine, "Offline meta-reinforcement learning with online self-supervision," in *International Conference on Machine Learning*. PMLR, 2022, pp. 17 811–17 829.

[296] M. Kocaoglu, C. Snyder, A. G. Dimakis, and S. Vishwanath, "CausalGAN: Learning causal implicit generative models with adversarial training," in *International Conference on Learning Representations*, 2018.

[297] R. She, P. Fan, X.-Y. Liu, and X. Wang, "Interpretable generative adversarial networks with exponential function," *IEEE Transactions on Signal Processing*, 2021.

[298] R. S. Lindsay, T. Funahashi, R. L. Hanson, Y. Matsuzawa, S. Tanaka, P. A. Tataranni, W. C. Knowler, and J. Krakoff, "Adiponectin and development of type 2 diabetes in the Pima Indian population," *The Lancet*, vol. 360, no. 9326, pp. 57–58, 2002.

[299] R. Miotto, L. Li, B. A. Kidd, and J. T. Dudley, "Deep patient: an unsupervised representation to predict the future of patients from the electronic health records," *Scientific Reports*, vol. 6, no. 1, pp. 1–10, 2016.

[300] T. Pham, T. Tran, D. Phung, and S. Venkatesh, "Predicting healthcare trajectories from medical records: A deep learning approach," *Journal of Biomedical Informatics*, vol. 69, pp. 218–229, 2017.

[301] S. Ramesh, H. Balaji, N. C. S. Iyengar, and R. D. Caytiles, "Optimal predictive analytics of Pima diabetics using deep learning," *International Journal of Database Theory and Application*, vol. 10, no. 9, pp. 47–62, 2017.

[302] S. Lekha and M. Suchetha, "Real-time non-invasive detection and classification of diabetes using modified convolution neural network," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 5, pp. 1630–1636, 2017.

[303] G. Swapna, R. Vinayakumar, and K. Soman, "Diabetes detection using deep learning algorithms," *ICT Express*, vol. 4, no. 4, pp. 243–246, 2018.

[304] A. Ashiquzzaman, A. K. Tushar, M. R. Islam, D. Shon, K. Im, J.-H. Park, D.-S. Lim, and J. Kim, "Reduction of overfitting in diabetes prediction using deep learning neural network," in *IT Convergence and Security 2017*. Springer, 2018, pp. 35–43.

[305] S. Spänig, A. Emberger-Klein, J.-P. Sowa, A. Canbay, K. Menrad, and D. Heider, "The virtual doctor: An interactive clinical-decision-support system based on deep learning for non-invasive prediction of diabetes," *Artificial Intelligence in Medicine*, vol. 100, p. 101706, 2019.

[306] B. P. Nguyen, H. N. Pham, H. Tran, N. Nghiem, Q. H. Nguyen, T. T. Do, C. T. Tran, and C. R. Simpson, "Predicting the onset of type 2 diabetes using wide and deep learning with electronic health records," *Computer Methods and Programs in Biomedicine*, vol. 182, p. 105055, 2019.

[307] K. Kannadasan, D. R. Edla, and V. Kuppili, “Type 2 diabetes data classification using stacked autoencoders in deep neural networks,” *Clinical Epidemiology and Global Health*, vol. 7, no. 4, pp. 530–535, 2019.

[308] P. Prabhu and S. Selvabharathi, “Deep belief neural network model for prediction of diabetes mellitus,” in *3rd International Conference on Imaging, Signal Processing and Communication (ICISPC)*. IEEE, 2019, pp. 138–142.

[309] K. S. Ryu, S. W. Lee, E. Batbaatar, J. W. Lee, K. S. Choi, and H. S. Cha, “A deep learning model for estimation of patients with undiagnosed diabetes,” *Applied Sciences*, vol. 10, no. 1, p. 421, 2020.

[310] M. D. Abràmoff, Y. Lou, A. Erginay, W. Clarida, R. Amelon, J. C. Folk, and M. Niemeijer, “Improved automated detection of diabetic retinopathy on a publicly available dataset through integration of deep learning,” *Investigative Ophthalmology & Visual Science*, vol. 57, no. 13, pp. 5200–5206, 2016.

[311] V. Gulshan, L. Peng, M. Coram, M. C. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros *et al.*, “Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs,” *JAMA*, vol. 316, no. 22, pp. 2402–2410, 2016.

[312] A. ElTanboly, M. Ismail, A. Shalaby, A. Switala, A. El-Baz, S. Schaal, G. Gimel’farb, and M. El-Azab, “A computer-aided diagnostic system for detecting diabetic retinopathy in optical coherence tomography images,” *Medical Physics*, vol. 44, no. 3, pp. 914–923, 2017.

[313] R. Gargeya and T. Leng, “Automated identification of diabetic retinopathy using deep learning,” *Ophthalmology*, vol. 124, no. 7, pp. 962–969, 2017.

[314] D. S. W. Ting, C. Y.-L. Cheung, G. Lim, G. S. W. Tan, N. D. Quang, A. Gan, H. Hamzah, R. Garcia-Franco, I. Y. San Yeo, S. Y. Lee *et al.*, “Development and validation of a deep learning system for diabetic retinopathy and related eye diseases using retinal images from multiethnic populations with diabetes,” *JAMA*, vol. 318, no. 22, pp. 2211–2223, 2017.

[315] S. Keel, P. Y. Lee, J. Scheetz, Z. Li, M. A. Kotowicz, R. J. MacIsaac, and M. He, "Feasibility and patient acceptability of a novel artificial intelligence-based screening model for diabetic retinopathy at endocrinology outpatient services: a pilot study," *Scientific Reports*, vol. 8, no. 1, pp. 1–6, 2018.

[316] J. Krause, V. Gulshan, E. Rahimy, P. Karth, K. Widner, G. S. Corrado, L. Peng, and D. R. Webster, "Grader variability and the importance of reference standards for evaluating machine learning models for diabetic retinopathy," *Ophthalmology*, vol. 125, no. 8, pp. 1264–1272, 2018.

[317] S. Wan, Y. Liang, and Y. Zhang, "Deep convolutional neural networks for diabetic retinopathy detection by image classification," *Computers & Electrical Engineering*, vol. 72, pp. 274–282, 2018.

[318] P. Ruamviboonsuk, J. Krause, P. Chotcomwongse, R. Sayres, R. Raman, K. Widner, B. J. Campana, S. Phene, K. Hemarat, M. Tadarati *et al.*, "Deep learning versus human graders for classifying diabetic retinopathy severity in a nationwide screening program," *npj Digital Medicine*, vol. 2, no. 1, pp. 1–9, 2019.

[319] D. S. Ting, C. Y. Cheung, Q. Nguyen, C. Sabanayagam, G. Lim, Z. W. Lim, G. S. Tan, Y. Q. Soh, L. Schmetterer, Y. X. Wang *et al.*, "Deep learning in estimating prevalence and systemic risk factors for diabetic retinopathy: A multi-ethnic study," *npj Digital Medicine*, vol. 2, no. 1, pp. 1–8, 2019.

[320] F. Arcadu, F. Benmansour, A. Maunz, J. Willis, Z. Haskova, and M. Prunotto, "Deep learning algorithm predicts diabetic retinopathy progression in individual patients," *npj Digital Medicine*, vol. 2, no. 1, pp. 1–9, 2019.

[321] I. Wittler, X. Liu, and A. Dong, "Deep learning enabled predicting modeling of mortality of diabetes mellitus patients," in *The Practice and Experience in Advanced Research Computing on Rise of the Machines (learning)*, 2019, pp. 1–6.

[322] T. Yamada, K. Iwasaki, S. Maedera, K. Ito, T. Takeshima, H. Noma, and N. Shojima, "Myocardial infarction in type 2 diabetes using sodium-glucose cotransporter-2

inhibitors, dipeptidyl peptidase-4 inhibitors, or glucagon-like peptide-1 receptor agonists: Proportional hazards analysis by deep neural network-based machine learning," *Current Medical Research and Opinion*, vol. 36, no. 3, pp. 403–409, 2020.

[323] I. Cruz-Vega, D. Hernandez-Contreras, H. Peregrina-Barreto, J. d. J. Rangel-Magdaleno, and J. M. Ramirez-Cortes, "Deep learning classification for diabetic foot thermograms," *Sensors*, vol. 20, no. 6, p. 1762, 2020.

[324] B. M. Williams, D. Borroni, R. Liu, Y. Zhao, J. Zhang, J. Lim, B. Ma, V. Romano, H. Qi, M. Ferdousi *et al.*, "An artificial intelligence-based deep learning algorithm for the diagnosis of diabetic neuropathy using corneal confocal microscopy: A development and validation study," *Diabetologia*, vol. 63, no. 2, pp. 419–430, 2020.

[325] J. W. Yau, S. L. Rogers, R. Kawasaki, E. L. Lamoureux, J. W. Kowalski, T. Bek, S.-J. Chen, J. M. Dekker, A. Fletcher, J. Grauslund *et al.*, "Global prevalence and major risk factors of diabetic retinopathy," *Diabetes Care*, vol. 35, no. 3, pp. 556–564, 2012.

[326] A. Grzybowski, P. Brona, G. Lim, P. Ruamviboonsuk, G. S. Tan, M. Abramoff, and D. S. Ting, "Artificial intelligence for diabetic retinopathy screening: a review," *Eye*, pp. 1–10, 2019.

[327] A. E. Johnson, T. J. Pollard, L. Shen, H. L. Li-wei, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. A. Celi, and R. G. Mark, "MIMIC-III, a freely accessible critical care database," *Scientific Data*, vol. 3, p. 160035, 2016.

[328] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 6765–6816, 2017.

Appendix A

Supplementary Information: Chapter 2

A.1 Literature Review on Diagnosis of Diabetes

The early diagnosis of diabetes can effectively improve the medical care and treatment for people living with diabetes. The standard diagnosis to confirm diabetes in clinics requires repeated glucose-based tests on hemoglobin A1c (HbA1c) and corresponding diagnostic criteria for different diabetes types [21]. However, due to the huge population and shortage of physicians in rural areas, the number of undiagnosed cases is significant and projected to increase in the future [4]. There is a high risk of developing diabetes without the onset of symptoms, especially for people with T2D, which could lead to long-term dysfunction of various organs and chronic complications.

Therefore, the need to detect onset diabetes or predict the diabetes risk arises, e.g. population screening and non-invasive systems. Table A.1 presents the current efforts at developing deep learning decision-support algorithms for the diagnosis of diabetes. In particular, various supervised and unsupervised learning approaches strategies have been applied, where DMLP models are the most widely employed. The feed-forward structures and simple connections make DMLP a good option for a binary classifier on EHRs, while AEs and RBMs are used to extract underlying patterns of the data without supervision. It is noted that many studies have used the publicly available dataset called Pima Indian Diabetes (PID), which It contains 768 instances with eight attributes and a binary label

Table A.1: Literature review on diabetes diagnosis

Ref.	Cases	Models	Data Sources	Development Process	Main Outcomes	Baselines
[299]	Classification of diabetes‡	Denosing AE	Mount Sinai Data Warehouse* (ICD-9)	Normalization; pre-process to obtain raw features; the data of training, validation and testing: 704,587, 5000, 76,214 patients	AUC: 0.907	Original descriptors, PCA (0.861)
[300]	Prediction of diabetes‡	Modified LSTM, attention pooling layer	An EHR dataset from a regional hospital (7191 patients, ICD-10)	The split for training, validation and testing: 2/3, 1/6 and 1/6 from 53,208 admissions	Precision of diagnosis, intervention, unplanned readmission: 66.2%, 78.7%, 79.0%	SVM, RF, plain RNN, LSTM (65.7%, 78.2%, 75.9%)
[301]	Detection of diabetes†‡	RBM and RNN	PID dataset from UCI repository$\star$	Feature selection by RFs; min-max normalization; the ratio for training and testing data: 80%, 20%	Sensitivity and precision: 90.66%, 75%	N/A
[302]	Prediction of diabetes†‡	Modified 1-D CNN and FC layers	25 breath samples collected by MOS sensors with 1000-sec intervals	The data for training and testing: 15 samples, 10 samples; leave-one out cross-validation	AUC of T1D, T2D, healthy subjects: 0.9659, 0.9625, 0.9644	SVD, SVM, PCA
[303]	Detection of diabetes	5-layer CNN, LSTM, and SVM	ECG data sampled at 500 Hz with digital bandpass filtering and thresholding collected from 40 people	HRV data from 71 ECG datasets (each contains 1000 samples); 5 fold cross-validation	Validation accuracy: 95.7%	Previous work using HRV
[304]	Detection of diabetes‡	DMLP with dropout	PID dataset from UCI repository*	The ratio of training and validation data: 90% and 10%	Accuracy: 88.41%	Previous work on the same dataset
[305]	Prediction of diabetes‡	DMLP	A population dataset (4814 participants,the majority are overweight)	Data cleaning (imputing missing values with the median); the ratio of training and testing data: 80% and 20% from 656 T2D subjects	AUC without and with HbA1c: 0.703, 0.840	SVM (0.679,0.825)
[306]	Prediction of the onset T2D‡	DMLP and a linear model	Practice Fusion dataset (9948 patients, ICD-9)*	Feature extraction by grouping 1312 features; the ratio of training and validation data: 70%, testing data: 30%; 10-fold cross-validation	Sensitivity: 31.17%, AUC: 84.13%	RF (29.12%, 16.07%)
[307]	Detection of diabetes‡	2 layer AE and a softmax layer	PID dataset from UCI repository*	Training the layer one by one with previous output; fine-tuning by supervised learning	Sensitivity: 87.92%, specificity: 83.41%, accuracy: 86.26%	Previous work on the same dataset
[308]	Prediction of diabetes‡	DBN	PID dataset from UCI repository*	Min-max normalization; feature selection by PCA; pre-training for RBMs; supervised fine-tuning	Sensitivity: 100%, F1 score: 0.808	DT, LR, RF, SVM, NB (75.9%, 0.760)
[309]	Detection of undiagnosed diabetes‡	2 hidden layer DMLP with dropout	An EHR dataset from a national survey (31,098 subjects, 4 years)	Combining 2013-2016 datasets;selecting features by LR; the data of training and testing: 11456 and 4444 subjects	AUC: 80.11%	LR, KNN, SVM, AdaBoost, Gaussian NB, RF (79.05%)

(diabetic or non-diabetic). The Pima Indians have a higher prevalence of T2D than any population [298], making the PID dataset popular in machine learning research. An advantage of using this dataset with the same metrics is to easily compare the results with the previous work employing various machine learning methods.

Nevertheless, the major limitation of applying deep learning on the PID dataset is the small number of patients and attributes. To prove the generalization of DNNs, the trained models need to be validated on a large population dataset. To this end, Miott *et al.* proposed a framework, namely Deep Patient, using a stack of denoising AEs to learn the representations from a large-scale dataset. The achieved AUC for the diagnosis of diabetes classification was 0.907 [299]. A recent study by Ryu *et al.* also employed a large dataset with 11,456 subjects [309]. They used a DMLP model to screen undiagnosed diabetes and achieved an AUC of 80.11% to detect undiagnosed diabetes. It is worth highlighting the data pre-processing step employed in these studies to extract related descriptors from the attributes of the patients, such as feature analysis and data normalization.

Moreover, the non-invasive detection of diabetes is emerging in several studies. Lekha *et al.* proposed a one-dimensional (1-D) CNN architecture to analyze the biomarkers in real-time breath signals for diabetes detection and classification [302]. The breath samples were collected by MOS sensors to analyze volatile organic compounds. The sensor array measured the content in a small gas chamber with an interval of 1000 seconds. Then the CNN classifier further processed these signals, which can reduce the need for feature selection and optimised the overall performance, compared to PCA, SVM, and singular value decomposition algorithm (SVD). In [303], the heart rate variability (HRV) from ECG was used as a biomarker to detect diabetes. The data was collected from a group of 40 people over a 10-minute duration. The ECG signals were sampled at 500 Hz with digital bandpass filtering and thresholding operations to reduce noise during the real-time detection. After deriving the heart rate time with the Pan-Tompkins algorithm, the study developed a hybrid deep learning model with CNN, LSTM, and SVM, and achieved a validation accuracy of 95.7%.

A.2 Literature Review on Diagnosis of Diabetes-Related Complications

In this category, most research has focused on the analysis of medical imaging to detect and diagnose multiple complications associated to people living with diabetes, as shown in Table A.2. Diabetes-related complications are diverse and regular examinations and clinical visits are time-consuming, expensive, and subjective [26]. For a long-term chronic disorder such as diabetes, its treatment is a heavy burden on the healthcare systems. Therefore, automated systems that are able to screen, detect, predict, and diagnose diabetes-related complications play an important role in population-based surveillance and monitoring.

Diabetic Retinopathy (DR) is the leading cause of vision impairments and blindness

Table A.2: Literature review on diagnosis of complications

Ref.	Cases	Models	Data Sources	Development Process	Main Outcomes	Baselines
[310]	Referable DR detection	CNN (Inspired by AlexNet, VGGNet)	(1) EyeCheck project, (2) Messidor-2 dataset*	Training data: 10,000 to 1,250,000 unique samples (1), testing data: 1748 images (2)	Sensitivity: 96.8%, specificity: 87.0%, AUC: 0.980	Classical detector by LR (AUC: 0.955))
[311]	Referable DR detection	CNN (Inception-v3), an ensemble of 10 networks	(1) EyePACS-1 dataset, (2) Messidor-2 dataset*	Batch normalization; pre-initialization by the ImageNet data; training and validation data : 80% and 20% of 128,125 retinal images, testing data: 4497 images (1) and 1748 images (2)	Sensitivity: 97.5% (1), 96.1% (2), specificity: 93.9% (1), 93.4% (2), AUC: 0.990 (1), 0.991 (2)	N/A
[312]	DR detection‡	A stack of non-negativity-constrained AEs	Images from 52 clinical scans with 12 retinal layers	Feature extraction by cumulative distribution function; training data: 40 subjects, testing data: 12 subjects	Sensitivity: 92%, specificity: 83%, accuracy: 100%	k-star (89%, 89%, 89%), KNN, RF, DT
[313]	DR detection	Customised CNN: (5 residual blocks), DT classifier	(1) EyePACS dataset, (2) Messidor-2 dataset*, (3) E-Ophtha dataset*	Feature extraction; training and validation data: 75,137 images (5-fold cross-validation, 1), testing data: 1368 images (2) and 405 images (3)	Sensitivity: 94% (1), 93% (2), 90% (3), specificity: 98% (1), 87% (2), 94% (3), AUC: 0.97 (1), 0.94 (2), 0.95 (3)	Previous work with machine learning on (2)
[314]	Referable DR detection	CNN (Adapted VGGNet)	Singapore national DR screening program	Each image was analyzed by two graders and one specialist, training data: 76,370 images (2010-2013 year), testing data: 71,896 images (2014-2015 year)	Sensitivity: 90.5%, specificity: 91.6%, AUC: 0.936	N/A
[315]	Referable DR detection	CNN (Inception-v3)	LabelMe dataset	21 grader validated the accuracy of the labels; training data: 58,790 images, cross-validation data: 8000 images	Sensitivity: 92.3%, specificity: 93.7%, 96% of participants satisfied with the model	Manual screening models
[316]	Moderate or worse DR detection	CNN (Inception-v4), an ensemble of 10 networks	EyePACS clinics, (2) Messidor-2 dataset*, (3) EyePACS-2 dataset	Gaussian process bandit algorithm (hyper-parameter tuning); training and validation data: 1,665,151 and 3737 images (1), (2), testing data: 1958 images (3)	Sensitivity: 97.1%, specificity: 92.3%, AUC: 0.986	Three retinal Specialists (sensitivity: 83.8%)
[317]	DR detection	CNN (VGGNet-s)	Kaggle dataset*	Normalization schemes and data augmentation; non-local means denoising; 5-fold cross-validation, training and validation data: 35,126 images	Sensitivity: 86.47%, specificity: 97.43%, AUC: 0.9786, accuracy: 95.68%	AlexNet, ResNet, VGGNet-16, VGGNet-19 (specificity: 96.49%), GoogleNet,
[318]	Referable DR detection	CNN (Inception-v4), an ensemble of 10 networks	A large-scale population (13 health regions, 7517 patients)	A cascade of thresholds (hyper-parameter tuning); testing data: 25,326 images	Sensitivity: 96.8%, specificity: 95.6%, AUC: 0.987	13 human regional graders (sensitivity: 74%)
[319]	Referable DR detection	CNN(Adapted VGGNet)	A multi-ethnic, multi-site dataset (5 races, 18,912 patients)	Training and validation data: 76,370 and 8000 images, testing data: 93,293 images	The estimation of DR prevalence: 16.1%, the AUC for referable DR: 0.863, the time taken to diagnose: 10.4h, risk factor: 0.743	10 retinal specialists and 7 professional graders (prevalence:15.9%, time: 1554.8 h)
[320]	Estimation of DR severity scale	CNN pillars (Inception-v3) and RF	(1) Kaggle dataset*, (2) 2 large clinical trials (530 patients)	SHAP for feature selection; transfer learning (1); 5-fold cross-validation: 4781 images (2)	AUC at month 6, 12, 24: 0.68, 0.79, 0.77	Well-trained reading center experts
[321]	Prediction of mortality in ICU	1-D CNN and 2 FC layers	MIMIC-III dataset*	Feature analysis by importance; addressing imbalance classes; training and testing data: 70% and 30% from 9000 subjects	AUC: 0.885	ANN (0.792), RF, DT
[322]	Prediction of myocardial infarction‡	DMLP	American commercial health plan	Descriptive statistics analysis, confounding factor analysis; extracting 199,116 patients	AUC: 0.767, with hazard ratio: 0.81 and 0.63	LR (AUC: 0.760)
[323]	Classification of diabetic foot	Customised 9-layer CNN	Plantar thermogram database with 167 subjects*	Data augmentation; patch extraction; the ratio of training and validation data: 70% and 30%, 10-fold cross-validation	Sensitivity: 0.9167, AUC: 0.8533	SVM, ANN, AlexNet, GoogLeNet
[324]	Detection of diabetic neuropathy†‡	U-Net CNNs (5 ensembles)	(1) BioImLab dataset*, (2) Beijing dataset, (3) ENA dataset	Patch extraction; training set: 1698 images (2), testing set: 2137 images (1), (3)	Fibre length 0.933, length/segment: 0.656, branch points: 0.891, nail points: 0.623	ACCMetrics model (0.825, 0.325, 0.570, 0.257)

in the world [325], which is often difficult to be detected until vision-threatening events occur. Fortunately, the state-of-art technologies of deep learning have shown great potential to meet this challenge and provide solutions to various DR problems reaching, in some cases, superhuman performance [326]. Following the success in the computer version, a large number of CNN-based models have been adopted to extract the representation from retinal fundus photographs. In Table A.2, nearly all the selected studies used CNNs to detect DR (10/11, 91%). The exception is the study by ElTanboly *et al.* which designed a multistage deep fusion classification network with a stack of non-negativity-constrained AEs to detect DR in optical coherence tomography (OCT) images for the patients who have almost normal retina appearances [312]. The AE model achieved high classification accuracy on an experiment with 52 subjects. As for CNN-based studies, most of the approaches are adapted or inspired from two popular architectures in the CV; VGGNet (4/10, 40%) and Inception (5/10, 50%). In [317], multiple popular CNN configurations were explored on the Kaggle dataset, where VGGNet-s obtained the highest accuracy in the experiments. Both of the architectures achieved satisfactory performance on DR detection. Abràmoff *et al.* proposed a VGGNet-based model to detect multiple classes of DR on Messidor-2 dataset [310], achieving a high sensitivity of 96.8% on referable DR. Then the VGG adapted architecture was validated in two large scale datasets with multi-ethnic populations [314, 319]. Their studies indicated deep learning methods can detect the referable DR with high accuracy but with much less time than human assessors. Gulshan *et al.* used an Inception-based architecture to detect referable DR and achieved the sensitivity of 96.8% and the specificity of 87.0%. The clinical settings to implement such systems were further investigated, including the feasibility and acceptability of outpatient settings [315] and grader variability [316]. Ruamviboonsuk *et al.* conducted a nationwide experiment to validate an Inception-based model [318]. Compared with human specialists, the deep learning model obtained significantly higher sensitivity and slightly lower specificity. Meanwhile, Arcadu *et al.* proposed an Inception model to predict DR progression by leveraging individual color fundus photographs [320].

The deep learning applications to other complications are also noted in the literature. Wittler *et al.* designed a CNN-based model to predict mortality based on the data from the intensive care unit (ICU) patients, achieving an AUC score of 0.885 [321]. The ICU dataset in this study, referred to as MIMIC-III, is freely accessible [327]. Williams *et al.* [324] proposed a U-Net CNN to quantify the nerve fiber properties in the diagnosis of diabetic neuropathy. Their results show an excellent localization performance for the quantification and the potential to be adopted in clinical settings. In [323], a customised CNN was designed to detect plantar ulcers on the thermography of diabetic foot. Moreover, Yamada *et al.* investigated the incidence of cardiovascular disease among three anti-diabetic drugs, using a DMLP model that achieved better results than conventional LR analysis [322].

Appendix B

Supplementary Information: Chapter 3

B.1 Notations

Table B.1 summarises the notations in formulation.

B.2 Data Split

Fig. B.1 shows the process of data split.

B.3 Hyperparameters

We employed the Hyperband algorithm [328] to speed up the hyperparameter optimization in the search space using the library of Keras Tuner. Table B.2 lists the hyperparameters used in this work.

B.4 Demographic and Clinical Characteristics

Table B.4 shows the demographic characteristics (Median (IQR)) and relevant clinical information (Mean±STD) for the employed datasets.

Table B.1: Notation table

Variable	Definition
$\mathbf{X}, \hat{y}, y, \hat{G}, G$	Model input, model output, target label, estimated BG level, actual BG level
t, r, L, D, L_T	Timestep, PH timestep, length of input sequences, input feature dimension, total length of a testing set
$\mathbf{G}, \mathbf{M}, \mathbf{I}, f_N$	CGM sequence, meal sequence, insulin bolus sequence, min-max normalization function
$\mathbf{W}_z, \mathbf{W}_r, \mathbf{W}_h$	Weights for input in update gate, reset gate, and activation in GRU
$\mathbf{U}_z, \mathbf{U}_r, \mathbf{U}_h$	Weights for hidden state in update gate, reset gate, and candidate activation in GRU
$\mathbf{b}_z, \mathbf{b}_r, \mathbf{b}_h$	Bias in update gate, reset gate, and candidate activation
$\mathbf{h}', \hat{\mathbf{h}}$	Hidden state of cell function input, candidate activation in GRU
$\overrightarrow{\mathbf{h}}, \overleftarrow{\mathbf{h}}, \overline{\mathbf{h}}$	Hidden state of forward output, backward output, concatenated output in bidirectional RNN
$a_i, \mathbf{c}_t, \mathbf{W}_a, \mathbf{W}_m$	Attention weights, context vector, weights for alignment scores, weights for attentional output
$\gamma, \lambda, \alpha, \beta$	Parameters of the NIG distribution
$\mathcal{L}, \mathcal{L}^N, \mathcal{L}^R, k$	Total loss, NLL loss, regularization loss, proportion of regularization loss
u^a, u^e	Aleatoric uncertainty and epistemic uncertainty
θ_j, θ_m	Parameters of task-learning model and meta-model
η, ϵ	Learning rate of Adam optimiser, step size of SGD
$\mathcal{T}, N, \mathcal{L}_{B_{\mathcal{T}_j}}, f_\theta$	Task in meta-learning, number of tasks, loss of mini-batch, model inference with parameters θ

Table B.2: Hyperparameters of FCNN

Hyperparameter	Value
Batch size	32
Length of input sequences	12
Hidden units of GRU layers	[128,64,32]
Hidden units of the attention layers	128
Regularization factor k	0.01
Adam learning rate in fine-tuning	1×10^{-4}
Adam learning rate η in task learning	1×10^{-3}
SGD learning rate ϵ	0.01
Number of meta iterations	2×10^4
Number of epochs	500
Early stopping patience	50

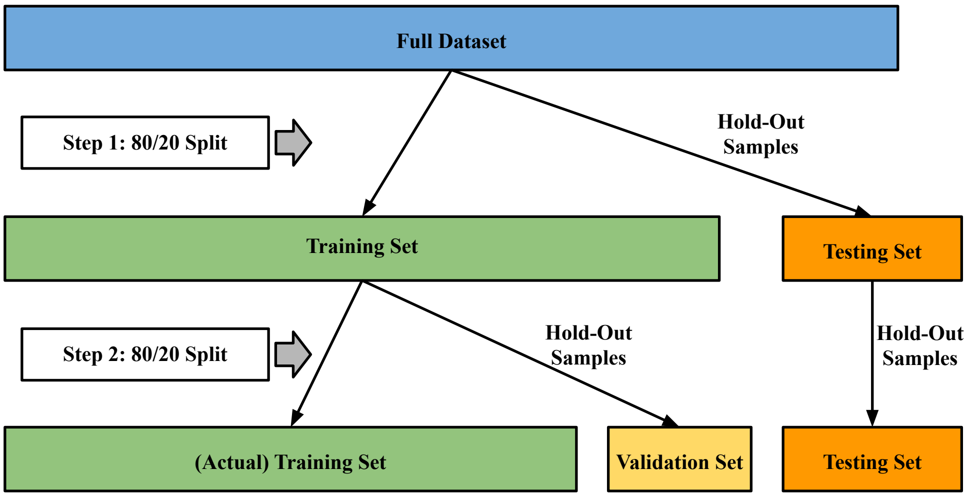


Figure B.1: Illustration of data split. A two-step hold-out split is used to divide full dataset into three parts: (actual) training, validation, and testing sets. The models were trained using the actual training sets, and validation and testing sets were unseen data. Then the validations sets were used for feature selection and hyperparameter tuning. Finally, the testing sets were used to provide unbiased evaluation and the prediction results reported in this work.

B.5 Additional Experimental Results

Table B.5 summarises the results of the glucose variability impact index (GVII) and glucose prediction consistency index (GPCI) for the ABC4D dataset with the FCNN model. Figure B.2 depicts the seven-day period CGM and prediction trajectories with the ARISES model.

B.6 Data Collected in ARISES clinical trial

Table B.6 summarises the daily entries recorded in the clinical trial, and Table B.7 shows the detailed information. Table B.8 presents the features extracted from multi-model data.

Table B.3: Hyperparameters of ARISES

Hyperparameter	Value
Hidden units of the Bidirectional GRU layer	128
Hidden units of the GRU layer	128
Hidden units of the attention layer	128
Learning rate of meta-learning	1e-4
Learning rate of fine-tuning	1e-5
Batch size	64
Number of epochs	500
Early stopping patience	50
Length of sequence	12
Feature Dimension	6

Table B.4: Demographic characteristics (Median (IQR)) and relevant clinical information (Mean±STD) of the T1D subjects in the OhioT1DM Dataset, ARISES, and ABC4D datasets

	OhioT1DM	ARISES	ABC4D
Demographic			
Age	50.0 (40.0-60.0)*	40.0 (30.0-49.0)	36.0 (29.0–46.0)
Gender (female/male)	5/7 (41.7% female)	6/6 (50.0% female)	15/25 (60.0% female)
HbA1c (mmol/mol)	N/A	50.4 (41.5-57.5)	61.0 (52.0–66.0)
Insulin regime (CSII/MDI)	12/0 (100.0 % CSII)	6/6 (50.0% CSII)	8/17 (47.1% CSII)
Clinical			
Mean BG level (mg/dL)	159.37 ± 16.33	161.25 ± 26.02	158.10 ± 25.06
Median BG level (mg/dL)	152.58 ± 18.32	154.96 ± 26.97	151.37 ± 25.83
eA1c (%)	7.18 ± 0.57	7.25 ± 0.91	7.14 ± 0.87
Time below range (%)	3.30 ± 2.25	2.93 ± 1.93	4.97 ± 3.93
Time in range (%)	63.54 ± 9.70	63.29 ± 16.00	63.15 ± 15.50
Time above range (%)	33.16 ± 10.71	33.78 ± 17.01	31.88 ± 16.02
Low blood glucose index	0.88 ± 0.48	0.78 ± 0.46	1.34 ± 0.89
High blood glucose index	7.15 ± 2.45	7.59 ± 4.17	7.37 ± 3.94
Inter-day CV (%)	36.63 ± 3.70	35.14 ± 4.47	37.50 ± 5.98
Intra-day CV (%)	31.18 ± 3.56	30.84 ± 4.78	31.86 ± 5.24

*Approximation for the de-identified data.

Table B.5: Glucose variability impact and glucose prediction consistency

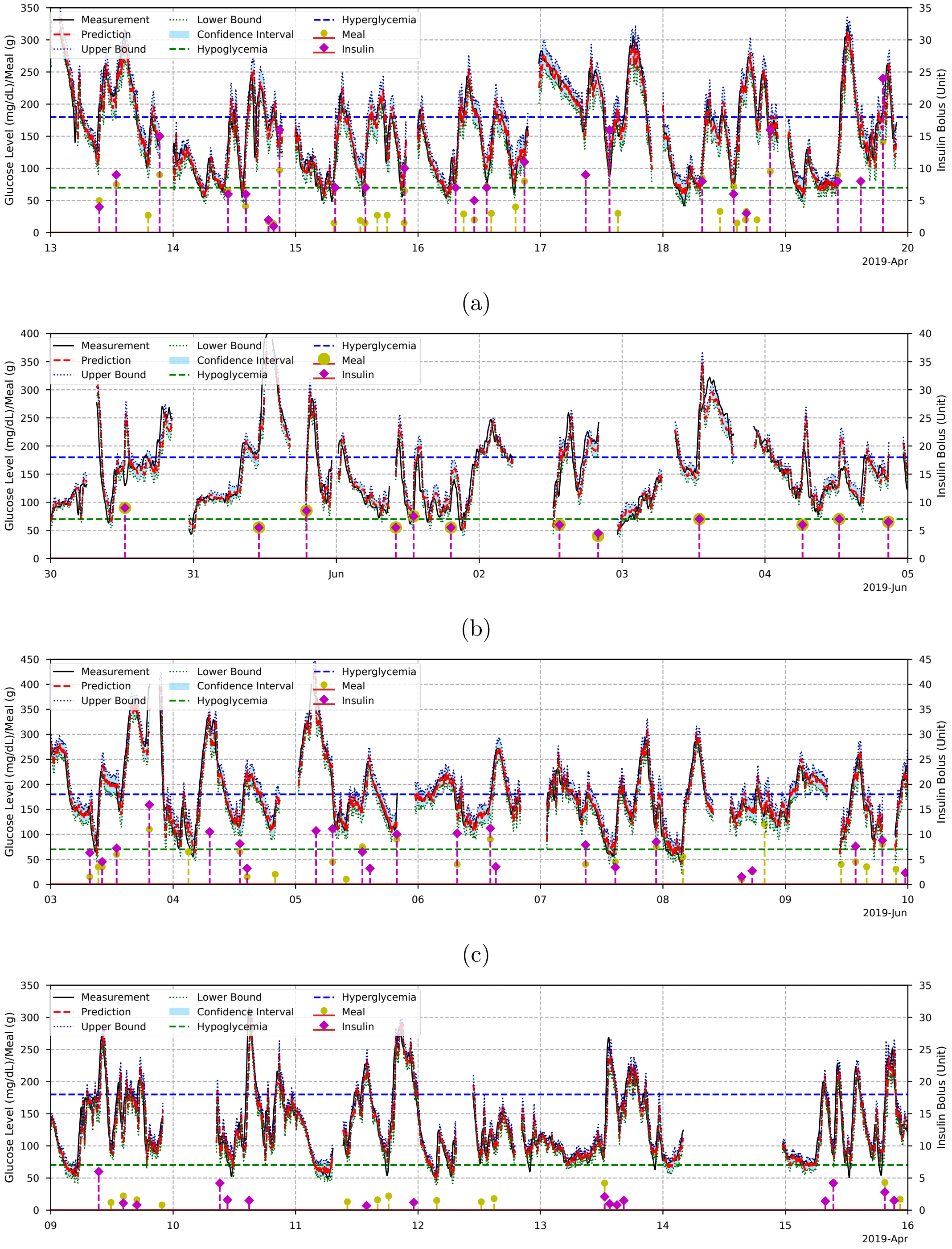
PH	Method	GVII	GPCI	r	p
30 min	FCNN	0.18	3.05	0.40	0.04
	CRNN	0.18	3.08	0.41	0.04
	Bi-LSTM	0.19	2.87	0.46	0.02
	Transformer	0.18	3.21	0.41	0.04
	SVR	0.22	2.70	0.55	< 0.01
	RFR	0.20	2.86	0.49	0.01
	ARIMA	0.16	3.20	0.37	0.07
60 min	FCNN	0.49	4.87	0.63	< 0.01
	CRNN	0.49	4.92	0.62	< 0.01
	Bi-LSTM	0.52	4.16	0.71	< 0.01
	Transformer	0.48	4.96	0.65	< 0.01
	SVR	0.51	4.70	0.65	< 0.01
	RFR	0.54	4.64	0.68	< 0.01
	ARIMA	0.51	5.50	0.59	< 0.01

Table B.6: Number of daily entries manually recorded by the participants in the phase I of the ARISES clinical study

Entries	**All (n=5767)**		**MDI (n=2988)**		**CSII (n=2779)**	
	% (n)	Median (IQR)	% (n)	Median (IQR)	% (n)	Median (IQR)
Carbohydrates	39.6% (2285)	207 (122-249.5)	41.2% (1233)	172 (127.5-285.5)	37.9% (1052)	213 (107.5-242)
Protein	16.9% (976)	132 (33-163)	17.5% (522)	42 (4-146)	16.3% (454)	158 (138-168)
Fat	16.5% (952)	135 (31-163)	16.5% (492)	40 (4-146)	16.5% (460)	158 (140.5-168.5)
Insulin Bolus	20.6% (1189)	111 (57-142.5)	18.9% (565)	101 (50-131)	22.5% (624)	123 (77.5-144.5)
Exercise	4.5% (262)	19.5 (13-29.5)	4.9% (148)	19.5 (13.8-28.3)	4.1% (114)	18.5 (13.5-27.3)
Alcohol	1.2% (72)	6 (4-9)	0.5% (14)	2.5 (0.75-5.25)	2.1% (58)	8 (6-17)
Stress	0.3% (18)	2.5 (2-3)	0.1% (4)	2 (2-2)	0.5% (14)	3 (2.5-4)
Illness	0.3% (18)	3 (3-4.8)	0.3% (10)	10 (10-10)	0.3% (8)	3 (2.5-3)

Table B.7: Information of daily entries manually recorded by the participants (Median (IQR)) in the phase I of the ARISES clinical study

Entries	**All**	**Baseline**	**Endpoint**	p
Total daily carbohydrate (grams)	160 (102-220)	160 (100-228)	145 (101-220)	0.77
Breakfast carbohydrate (grams)	41 (25-60)	40 (26-60)	42 (29.3-58.8)	0.68
Lunch carbohydrate (grams)	45 (30-60)	50 (33.8-62.8)	40 (30-65)	0.14
Dinner carbohydrate (grams)	50 (40-70)	50 (40-70)	50 (40-75)	0.50
Daily Bolus insulin (units)	20 (14.8-28)	20.2 (15.9-28.6)	20 (14-27.5)	0.29
Number exercise	19.5 (11.8-30.3)	8.5 (7.8-12.3)	7 (2.3-8)	0.10
Daily engagement	7 (4-9)	7 (4.8-9)	6 (4-8)	0.08
Number of interactions	396 (237-732.3)	152 (121-255.3)	127 (89-229.8)	0.42



(a)

(b)

(c)

(d)

Figure B.2: Seven-day period CGM and prediction trajectories of T1D adults over a 30-minute prediction horizon, each representing one of the four subgroups in the clinical study. (a) a male participant with MDI regimen. (b) a female participant with MDI regimen. (c) a male participant with CSII regimen. (d) a female participant with CSII regimen.

Table B.8: Description of features

Wristband (Empatica E4)	
IBI	Inter-beat intervals
meanHR	Mean heart rate
medianNNI	median value of normal-to-normal (NN) intervals
SDNN	Standard deviation of NN intervals
RMSSD	Root mean square of successive differences between adjacent NNs
CVSD	Coefficient of variation of successive differences
CVNNI	Coefficient of variation of NN intervals
pNNX	Percentage of successive NN intervals greater than 50 ms
LHR	Low-/high-frequency power ratio
EDA	electrodermal activity
SCL	Skin conductance level
SCR	Skin conductance response
ACC	Average 3-D acceleration
TEMP	Skin temperature

CGM (Dexcome G6)	
CGM	CGM measurement sequence
timeIndex	Normalised time sequence

Daily entries	
Carb	Amount of carbohydrate intake
IOB	Insulin on board
Bolus	Amount of insulin bolus delivery
COB	Carbohydrate on board

Appendix C

Supplementary Information: Chapter 4

C.1 Reward Functions in Basal Glucose Control

Table C.1 shows the results of different reward functions explored in the experiments.

Table C.1: Reward functions and corresponding scores

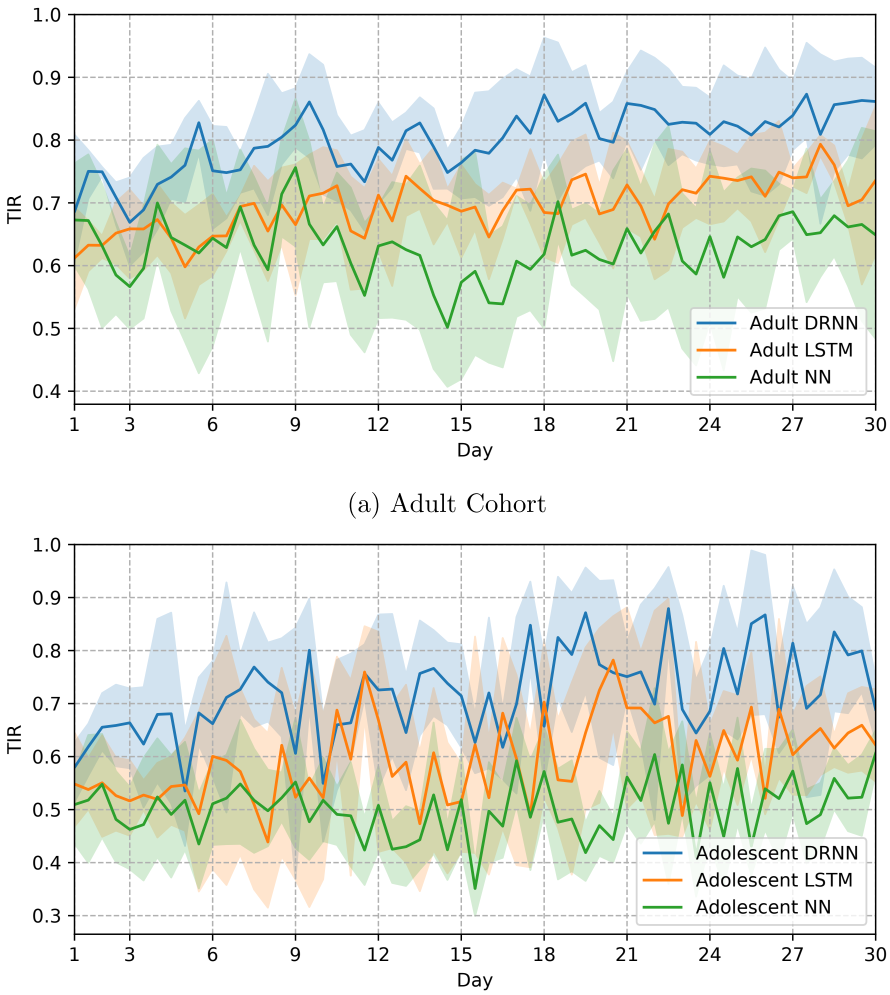
BG Range (mg/dL)	Reward Scheme 1	Reward Scheme 2	Reward Scheme 3	Reward Scheme 4
0-30	-10	-1	-1	-1
30-70	-1	-0.5	$-0.5 + \frac{GL-70}{80}$	$-0.6 + \frac{GL-70}{100}$
70-90	+0.1	+0.1	+0.1	+0.1
90-140	+1	+1	+1	+1
140-180	+0.1	+0.1	+0.1	+0.1
180-300	-1	-0.5	$-0.5 - \frac{GL-180}{240}$	$-0.4 - \frac{GL-180}{200}$
> 300	-10	-1	-1	-1
TIR Score (%)	75	88	86	93

C.2 DNN Selection in Basal Glucose Control

Figure C.1 shows the TIR results achieved with the different neural network architectures that we evaluated in the experiments. Considering that the input data is a multi-dimensional time-aligned sequence, we assumed that an RNN-based model would be a good candidate to map the multiple-step historical data. Therefore, we explored conventional LSTM, DNNs with five fully-connected layers and DRNNs as the potential structure of DQNs. The LSTM architecture has recently achieved great success in time-aligned tasks, but in our case, it obtains lower TIR results than the DRNN. DMLP is commonly used in DQNs as a basic structure. However, the DMLP curve in Fig. C.1 shows large variability and lower mean TIR. Less variability indicates a better capability to account for within-subject variability. Thus, the DMLP structure was disregarded. Regarding the DRNNs, the generalised model achieves a good initial performance at the beginning of personalised training. In addition, the DRNN curve has a positive trend and small variability, which indicates its effectiveness at adjusting the models for a specific subject through a short period of time. Finally, DRNN prediction models ranked top in Blood Glucose Level Prediction Challenge in 2018 [128], outperforming various DNNs (e.g. one-dimensional CNNs and bidirectional RNNs). Therefore, DRNNs were naturally selected as the DQN modules for this work.

C.3 Hyperparameters

In Table C.2, we list the hyperparameters that have been used for double DQN. For each parameter, we performed limited tuning based on the state-of-art DQN [80]. Table C.3 lists the hyperparameters of DDPG model for meal insulin bolus recommendation. Table C.3 shows the hyperparameters of offline DRL and OPE. All the parameters are identical across all the considered subjects.



(a) Adult Cohort

(b) Adolescent cohort

Figure C.1: TIR results (mean, 95% CI) corresponding to DRL-DH during the personalised training for the adult and adolescent cohorts. The blue, orange and green lines show the results of DRNN, LSTM and DMLP models, respectively.

Table C.2: Hyperparameters of double DQN

Hyperparameter	Value
Exploration before learning k	2000
Generalised network update period T_G	1000
Generalised DQN ε-greedy	0.5→0.01
Personalised network update period T_P	100
Discount factor γ	0.9
Adam learning rate	1×10^{-5}
Batch size	32
Number of time steps L	12
Replay buff size $\mathcal{B}$	5000
Prioritization exponent α	0.3
Importance-sampling exponent β	0.4→1.0
Multi-step return λ_1	0.1
L2 regularization λ_2	1×10^{-5}
Cell type	Vanilla RNN
DRNN dilation	[1, 2, 4]
Hidden nodes of DRNN layers	[32, 64, 128]

Table C.3: Hyperparameters of DDPG

Hyperparameter	Value
The length of CGM measurements L	6
The hidden units of DNNs	[200, 200, 10]
The learning rate of the actor	0.0001
The learning rate of the critic	0.0001
The size of replay memory N	500
Batch size	32
Soft replacement τ	0.01
Target network update period $\mathcal{T}$	100
Discount factor γ	0.9
The degree of prioritization α	0.6
Compensation factor β	$0.4 \rightarrow 1$
Priority constant ϵ	0.00001

Table C.4: Hyperparameters of offline DRL and OPE

Hyperparameter	Value
Actor learning rates	5×10^{-5}
Batch size M	64
Critic learning rates	1×10^{-4}
Discount factor γ	0.97
Episode length L	72 (6 hours)
Hidden units of network layers	$[256, 256, 256]$
Interval to delay policy update t_d	2
Interval to update target networkst_u	100
Normalization constant σ	1×10^{-3}
Number of DRL training steps T_{DRL}	3×10^4
Number of OPE training steps T_{OPE}	2×10^4
Proportion of soft update μ	0.01
Regularization factor α	2.5

Appendix D

Supplementary Information: Chapter 5

D.1 Hyperparameters

Table D.1 lists the hyperparameters of GluGAN.

Table D.1: Hyperparameters of GluGAN

Hyperparameter	Value
Length of glucose time series L	24
Iterations of embedding learning $\mathcal{T}_R$	10,000
Iterations of supervised learning $\mathcal{T}_S$	10,000
Iterations of joint learning $\mathcal{T}_J$	25,000
Ratio of reconstruction loss λ_1	1
Ratio of unsupervised loss λ_2	10
Number of the inner loop steps k	2
Batch size	128
Threshold of discriminator loss l_D	0.15
Hidden units of the RNN layer	64
Learning rate of the Adam optimiser	0.001
Early stopping patience	50

Appendix E

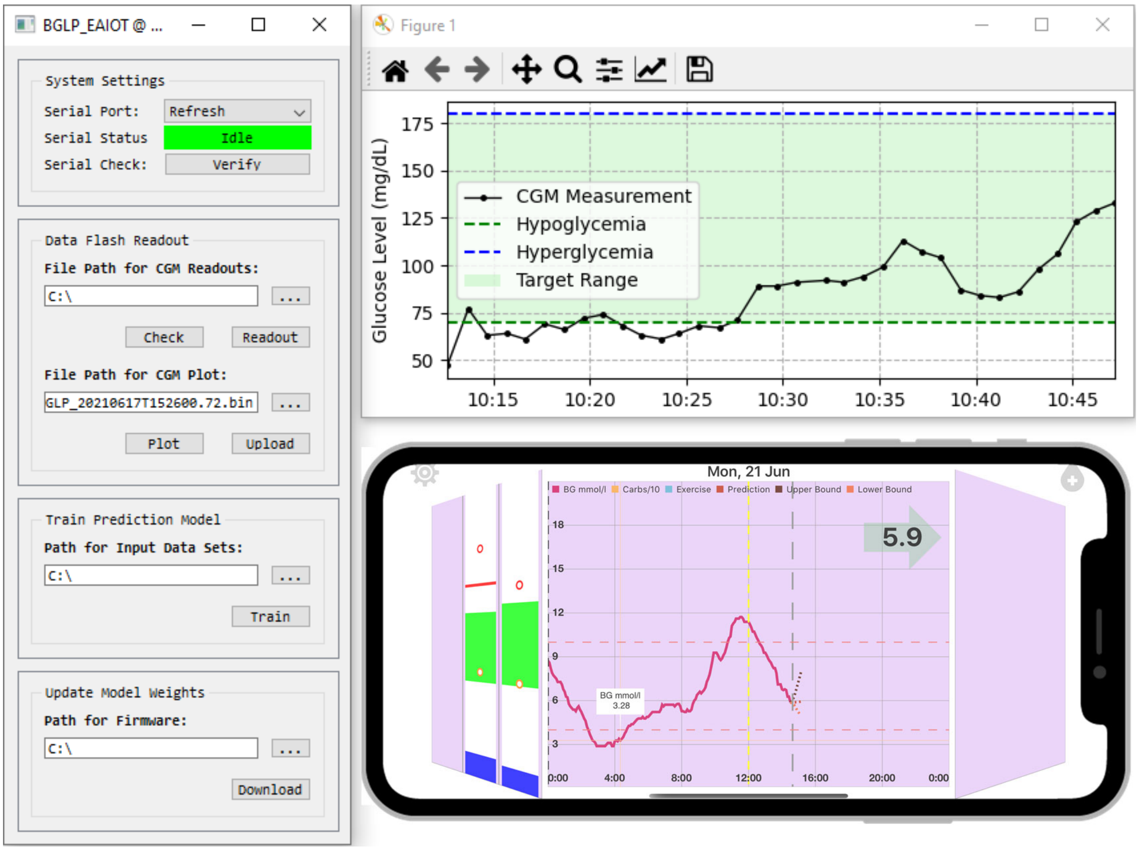
Supplementary Information: Chapter 6

E.1 GUIs of Smartphone and Desktop Platforms

Fig. E.1 depicts the GUIs of the iOS app and the desktop platform developed by Swift 4.2 and PyQt5 5.15, respectively. The desktop platform consists of multiple panels, including system settings, data readout, model training and update, and visualization of historical CGM data. Besides the historical trajectories, the smartphone platform also supports the visualization of the current CGM value and trend as shown by the green arrow.

E.2 Hyperparameters

Table E.1 listed the hyperparameters used in the E3NN model, which were determined by the Hyperband algorithm [328] with the Keras Tuner.



BGLP_EAIOT @ ...

System Settings

Serial Port: Refresh

Serial Status: Idle

Serial Check: Verify

Data Flash Readout

File Path for CGM Readouts: C:\

File Path for CGM Plot: GLP_20210617T152600.72.bin

Train Prediction Model

Path for Input Data Sets: C:\

Update Model Weights

Path for Firmware: C:\

Figure E.1: Overview of the smartphone and desktop GUIs.

Table E.1: Hyperparameters of E3NN

Hyperparameter	Value
Hidden units of GRU layers	[64,32]
Hidden units of the attention layer	64
Hidden units of the dense layer	64
Dropout rate	0.1
Learning rate	1×10^{-3}
Length of input sequences	12
Batch size	32
Number of epochs	300
Early stopping patience	30